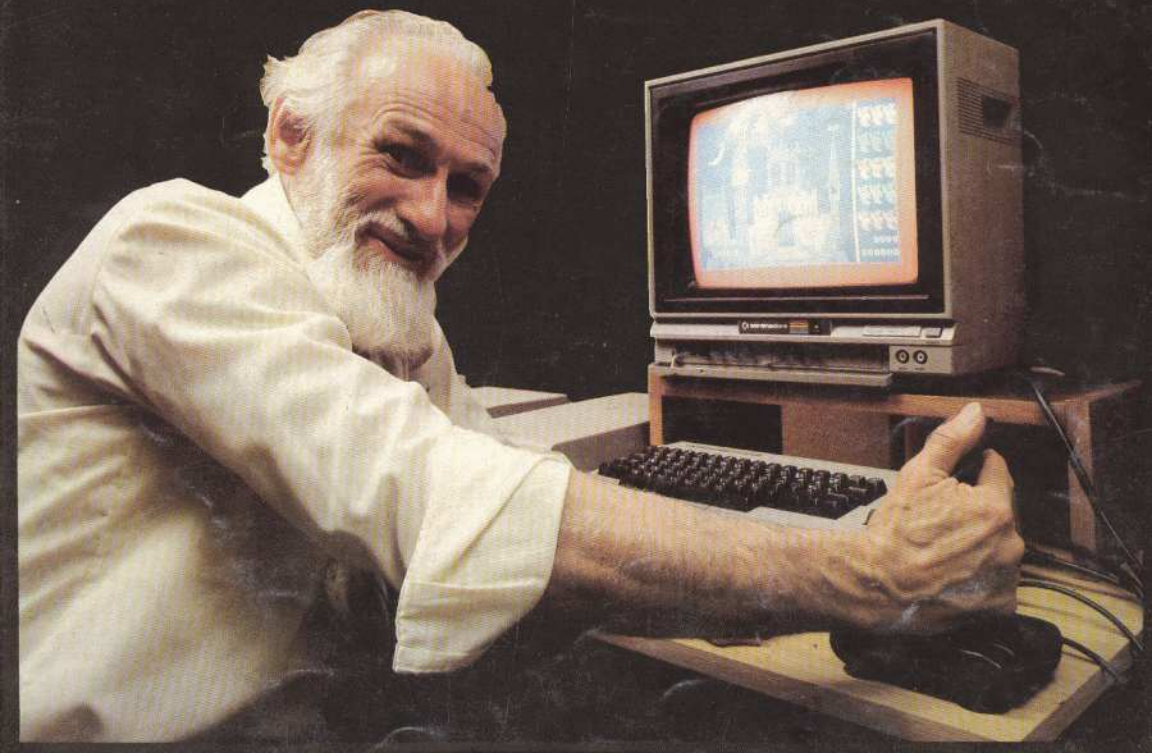
MIKROSZÁMÍTÓGÉP MAGAZIN

A NEUMANN JÁNOS SZÁMÍTÓGÉP-TUDOMÁNYI TÁRSASÁG LAPJA

1984/3



ÖKISZ
Szervezési
és Számítástechnikai Vállalat

MIKROSZÁMÍTÓGÉP MAGAZIN

A Neumann János Számítógéptudományi Társaság lapja

A kiadvány a Tudományszervezési és Informatikai Intézettel együttműködve készül

A szerkesztő bizottság vezetője:
Kovács Győző

Munkatársak:

Broczkó Péter (hírek)
Buday György István (személyi számítógépek)
Garádi János (agyafúrmány, rövid és ravasz programok)
Jakab Ágnes (ember–gép kapcsolat)
Kovács Győző (levelezés)
Lindner László (sakkprogramozás)
Nacsa Sándor (termékismertető)
Pataki Ernő (programozástechnika)
Petróczy Judit (könyvek)
Pogány Csaba (alkalmazástechnika, tanfolyam)
Simonyi Endre (klub)
Varga András (iskola – számítógép)
Vass Nándor (alkalmazások)
Votisky Zsuzsa (játékprogramok)

Felelős szerkesztő:
Könyves Tóth Pál

Szerkesztőség:
Budapest V., Báthori u. 16.
Telefon: 329-349, 329-390

Kiadja: a Lapkiadó Vállalat
Felelős kiadó:
Siklósi Norbert vezérigazgató
Kiadóhivatal: Budapest VII., Lenin krt. 9–11.
Postacím: 1906 Budapest, Pf. 223.
Telefon: 429-350, 221-285

Terjeszti a Magyar Posta
Előfizethető
bármely postahivatalban,
a kézbesítőknél,
a Posta hírlapüzleteiben
és a Posta
Központi Hírlap Irodában
(Budapest V.,
József nádor tér 1.,
postacím: 1900 Budapest)
közvetlenül
vagy postautalványon,
valamint átutalással
a PKHI 215-96162
pénzforgalmi jelzőszámra.
Előfizetési díj:
egy évre 168,– Ft,
fél évre 84,– Ft.

Szedte:
a Nyomdaipari Fényszedő Üzem (847481/09)

Nyomás:
Petőfi Nyomda, Kecskemét,
Külső Szegedi út 6.
(84.41779)
Telefon: 20466
Felelős vezető:
Ablaka István igazgató

INDEX: 25629

ISSN 0236-6088

Címképünk:
Commodore–64
Egy jó programmal játék az üzlet

NOVOTRADE

Tartalom

Számítástechnikai népoktatás	2
Adok – veszek – cserélek	27
Mi a régi az újban és mi a rossz a jóban?	34
Játék volt a javából!	38
SZEMÉLYI SZÁMÍTÓGÉPEK	
Mi a személyi számítógép?	3
ISKOLA – SZÁMÍTÓGÉP	
A személyi számítógépek hatása a családra, iskolára az USA-ban	5
Számítástechnika az oktatásban III.	6
Iskolaszámítógép-pályázat és sakkoktatás	8
Valószínűségelmélet a gyakorlatban	9
Iskolaszámítógép – BASIC – tanárok	10
TANFOLYAM	
Alapozás IV.	14
PROGRAMOZÁSTECHNIKA	
BASIC interpreterek – a program változói	16
TERMÉKISMERTETŐ	
Commodore–64	20
PIAC	
Táguló μvilág	22
Házi számítógép – Made in NDK	26
Bulgária – mikroperspektivában	27
Hazai szoftverkínálat mikroszámítógépekre 1984	29
SZÁZLÁBÚ	24
μKLUB	
Építsünk számítógépet! II.	32
Ismeretlen kódok	35
AGYAFÚRMÁNY	36
RÖVID ÉS RAVASZ PROGRAMOK	37
JÁTÉKPROGRAMOK	40
AZ OLVASÓ ÍRJA	42
SAKKPROGRAMOZÁS	44
HÍREK, ÉRDEKESSÉGEK	45
KÖNYVEK	47

Számítástechnikai népoktatás

„Én egész népemet fogom nem középiskolás fokon tanítani!"

(József Attila)

Ebben állapodtunk meg a Magyar Televízió és a SZÁMALK illetékeseivel. Ez az a program, amelynek keretében minden eddiginél szélesebb körben szeretnénk megismertetni a számítástechnikát.

A számítástechnikai képzés először a 70-es évek elején jelentett hazánkban különösen sürgős és óriási feladatot, amikor egyszerre nagy számú ESZR számítógép érkezett az országba, és egy-két év alatt 5–6000 szakembert – programozókat, szervezőket és karbantartó mérnököket – kellett üzemeltetésükhöz kiképezni. Akkor jött létre a SZÁMOK, amely oktatási licencek vásárlásával, lelkes oktatógárda segítségével sikeresen birkózott meg a feladattal: a lehető legrövidebb idő alatt kiképezte a legszükségesebb szakembereket.

Abban az időben az üzembe állított berendezések száma két év alatt háromszorosára emelkedett. Ezzel szemben az elmúlt három évben az országban gyártott és hivatalosan vagy magánúton behozott számítógépek száma – a személyi számítógépeket is beleértve – kb. tizenöt-hússzorosára emelkedett. Ma tehát már nem néhány ezer, hanem néhány tízezer – félve mondom, de talán százezer – embert kell a számítástechnikával megismertetni, a számítógépek alkalmazására kiképezni. Ezt – az előzőhöz nem is mérhető – feladatot már nyilvánvalóan csak tömegkommunikációs eszközökkel, egész pontosan a Televízióval együtt lehet megoldani.

Az említett szerződés szerint az Iskolatelevízió készíti a sorozatot, a SZÁMALK biztosítja az előadót, elkészíti a programozott tankönyv kéziratát és vizsgáztat, az NJSZT és a μ M pedig megszervezi a tanuláshoz a megfelelő számítógépes hátteret.

Az első tanfolyam a tervek szerint a jövő év elején indul, és célja a BASIC nyelvű programozás megtanítása. Nem kell a számítástechnikához különösen érteni annak belátásához, hogy programozást tanulni gyakorlás nélkül – fából vaskarika. Az is nyilvánvaló, hogy a következő öt hónapban nem lehet az ország valamennyi településére számítógépeket küldeni.

Ezért a Tudományszervezési és Informatikai Intézettel együtt felhívással fordulunk az ország valamennyi intézményéhez,

de a személyi számítógéppel rendelkező magánosokhoz is, hogy csatlakozzanak **μ Klub** mozgalmunkhoz, alakítsanak **μ klubokat** a számítástechnikai népoktatás támogatására.

μ klub lehet szoftverházak, gyárak, iskolák, művelődési házak, tanácsok és más intézmények számítógépterme, esetleg baráti körök vagy magánszemélyek számítógépe. A **μ klubhoz** az az intézmény tartozik, amely megengedi bárkinek, aki előzetesen jelentkezett, hogy zavartalanul leüljön egy (főleg személyi) számítógéphez vagy egy terminálhoz, és ott gyakorló programot futtasson, sőt szaktanácsot is adnak lelkes számítástechnikai szakemberek segítségével, akik hajlandók ezért az adott intézményben ügyeletet tartani. A **μ Klub** mozgalom a segítőkész intézmények és programozó szakemberek támogatására számít. A tanfolyam indulása előtt a **μ klubok** címét és a klubért felelősök nevét nyomtatott füzet formájában kiadjuk, hogy a tv-nézők a **μ klubokat** felkereshessék, illetve a számítógépek kipróbálására előzetesen jelentkezhessenek.

A μ M emblémát készít, amelyet szeretnénk, ha kiragasztanának a **μ klub** bejárata. A tv-tanfolyam végén vizsgát tett hallgatók szavazata alapján a legjobb **μ klubokat** és a legtöbbet segítő szakembereket megjutalmazzuk.

A számítástechnikai társadalomnak először nyílik alkalma arra, hogy ilyen nagyformátumú társadalmi akcióban részt vegyen, és minden eddiginél nagyobb lépést tegyen a számítástechnika elterjesztése érdekében. Gondolom, nem kell külön mondani, hogy a mozgalomból a szakma is sokat profitálhat. Sokat panaszkodunk, hogy a felhasználók igénye elmarad a számítástechnika által kínált lehetőségektől. Talán az a baj, hogy a nagyközönség nem tudja, mit kívánhat. Talán ez az alkalom hozza közel a nem számítástechnikai szakembert a számítógéphez, és adja meg azt a minimális ismeretet, amely elegendő ahhoz, hogy a munkahelyi alkalmazásra is megszülessen az igény.

Szeretnénk remélni, hogy nem hóbortos álmokat kergetünk – a tv-népoktatást a szakma közös sikerének könyvelhetjük el.

Várjuk és előre is köszönjük a **μ Klub** mozgalomban való részvételüket.

KOVÁCS GYŐZŐ

Mi a személyi számítógép?

A sorozat első két részében a személyi számítógépek meghatározásával, főbb jellemzőivel és hardverelemeivel foglalkoztam. Ezúttal a személyi számítógépek szoftveréről lesz szó.

A számítógépek hardvere önmagában nem alkalmas a felhasználó problémáinak megoldására. Ahhoz, hogy feladatokat tudjunk megoldani a személyi számítógéppel, szükség van szoftverre, amely programok hierarchikusan egymásra épülő rendszere. Kapcsolatot teremt a gépi kódban vezérelhető hardver és a felhasználó által megoldandó feladathoz illeszkedő kifejezési mód között. A szoftver két fő csoportra osztható: alapszoftverre és alkalmazói szoftverre.

Az alapszoftver

Az alapszoftver legfontosabb része az operációs rendszer. Ez áll a legközelebb a hardverhez. Biztosítja a gépi lehetőségek maximális kihasználását. Feladata az információáramlás megszervezése a felhasználói programok és a hardver megfelelő elemei között.

Az alapszoftverhez tartoznak a gép használatához alapvetően szükséges segédprogramok, mint például a mágneslemez-inicializáló programok, a másolóprogram, a rendező-válogató programok stb.

Az alapszoftver részét képezik továbbá a fordító- és az értelmező – interpreter – programok is, amelyek a megoldandó feladat megfogalmazását és a gép számára érthetővé tételét biztosítják.

Az első számítógépek megalkotásakor még nem álltak rendelkezésre fordítóprogramok. Ezeket a gépeket még ún. gépi kódban programozták. Minden gépi utasításnak egy bináris számsor felel meg. A feladatok megoldásának algoritmusát ilyen gépi kódok sorozatával kellett a számítógép számára érthetővé tenni. Ily módon még a legegyszerűbb probléma megoldása is nehéz, fáradságos munkát igényelt. Ezért hamarosan megjelent az első programozási nyelv, az Assembly, amelyben minden gépi kódnak egy könnyebben megjegyezhető angol szó vagy szimbólum felel meg. (Például a „tölts be az akkumulátorba" utasítás a LOADA, az „add hozzá az akkumulátorhoz" utasítás az ADD-dal írható le.) Ennél a nyelvnél minden adat elhelyezéséről és címzéséről a programozónak kell gondoskodnia.

A megírt Assembly nyelvű programot az assembler fordító alakítja át a számítógép által érthető gépi kódra. Minden számítógép – így a személyi számítógépek is – rendelkeznek assembler fordítóval. Mind a mai napig a legtöbb helyen ezen a nyelven írják a gépközeli, gyakran futó programokat, így az operációs rendszereket és egyéb alapszoftver-elemeket, bár már léteznek intelligensebb eszközök (CDL2, C nyelv stb.), amelyek közel azonos hatékonyságú programok írására alkalmasak.

Az idők folyamán újabb, egyre magasabb szintű programozási nyelvek jelentek meg. Ezeket két osztályba szokták sorolni: interpreterek (értelmezők) és compilerek (fordítók). Többnek közülük – mint például a BASIC-nek és a COBOL-nak – létezik mind interpreter, mind compiler változata.

Az interpreter futása közben a magas szintű nyelven megírt programot soronként fordítja le gépi kódra, és ezt azonnal végre is hajtatja a számítógéppel. A compiler viszont még a program végrehajtásának megkezdése előtt a teljes programot átalakítja gépi kódra. Az így először végrehajtható formájúvá – gépi kódúvá – átalakított program lényegesen gyorsabban fut, mint az interpreter által kezelt, mivel a fordítás nem a program futása közben történik.

Minél magasabb szintű egy programozási nyelv, annál közelebb áll az emberi gondolkodáshoz – annál könnyebb a kérdéses feladatot megfogalmazni vele –, és annál messzebb áll a gépi kódtól – vagyis annál kevésbé kell figyelembe venni a programozáskor a gép felépítését, működését. A magasabb szintű nyelvek használatakor például nem kell külön gondoskodni az adatok elhelyezéséről, címzéséről, nem kell tudni, hogy az egyes műveleteket – mondjuk az összeadást – hol és hogyan hajtja végre a számítógép. Egyszerűen le kell csak írni azt, hogy A = B+C, míg az Assembly nyelvvel dolgozónak a B adatot be kell töltenie az akkumulátorba arról a címről, ahol a tárolóban előzőleg elhelyezte, a C értéket hozzá kell adnia az akkumulátor tartalmához, majd az így nyert eredményt ki kell íratnia a tárolónak arra a helyére, ahol a C értékének előzőleg helyet biztosított.

Az első általánosan használt magasabb szintű programozási nyelv a FORTRAN volt. A személyi számítógépek legelterjedtebb nyelve a BASIC. Ma egyre kedveltebbé válik a PASCAL. Az utóbbi segítségével olyan programok írhatók, amelyek szerkezete világos, más számára is könnyen érthető. Jól alkalmazkodik a felhasználó egyéni stílusához. Igen nagy előnye, hogy szinte minden gépre – különösen a személyi számítógépekre – létezik PASCAL fordító. Lévén, hogy ezek nagyrészt az 1973 óta szabványként elfogadott PASCAL alapján készültek, és a szabványban definiált részt mind tartalmazzák – inkább csak újabb kiegészítő szolgáltatásaikban különböznek egymástól –, az egyszer egy gépre már megírt PASCAL program minimális munkával áttehető más gépekre is.

Igen nagy jelentőségük van az ilyen hordozható programoknak. Ne felejtsük el, hogy ma a számítógépek 4–5 év alatt erkölcsileg elavulnak. Ennyi idő alatt ugyanis olyan új gépeket fejlesztenek ki és hoznak piacra, amelyek mennyiségileg és minőségileg egyaránt többet nyújtanak elődeiknél, így a felhasználó gazdaságilag is érdekelt gépének lecserélésében. A szoftverkészítés technológiája az egyre intelligensebb programozási nyelvek és az egyre korszerűbb szoftvertechnológiák megjelenése ellenére sem tudott lépést tartani a hardvereszközök fejlődésével. Így állt elő az a helyzet, hogy egy teljes felhasználói rendszer árából a hardver kevesebbe kerül, mint a szoftver. Amennyiben egy alkalmazói szoftver megírásakor a hordozhatóság szempontjait figyelembe vették, az új gépen a régi szoftver legfeljebb minimális ráfordítás árán szintén használható. Ily módon a felhasználó mentesülhet a számára szükséges szoftver ismételt megvásárlása vagy elkészítése alól.

Egy probléma számítógéppel történő megoldásához két dolgot kell tudnunk: azt, hogy mit és hogyan akarunk végrehajtani. Az eddig tárgyalt hagyományos programozási nyelvekkel algoritmusokat lehet és kell megfogalmazni, vagyis azt, hogy a feladatot a számítógépnek hogyan kell megoldania.

Ismertek már, és az ötödik generációs számítógépek megjelenésével minden bizonnyal széles körben fognak elterjedni az ún. „nagyon magas szintű" vagy mások által „újelvű"-nek nevezett programozási nyelvek.

E nyelvek kialakításának legfőbb törekvése, hogy a programozónak egyre inkább a „mit"-tel és egyre kevésbé a „hogyan"-nal kelljen foglalkoznia, vagyis nem a feladat végrehajtási menetét, hanem csak magát a feladatot kell megfogalmaznia a gép számára.

Ezek közé a nyelvek közé tartozik a PROLOG, amely a tapasztalat szerint a számítógépek új alkalmazási területeit nyitja meg. A PROLOG-nak, amely a japán ötödik generációs számítógépek alapnyelve lesz, egy mini változata némelyik személyi számítógép felhasználóinak már ma rendelkezésére áll. A PROLOG segítségével például olyan problémák egyszerű és gyors megoldása vált lehetővé, mint a gyógyszerek egymásra hatásának vizsgálata. Hagyományos módszerekkel ezt a feladatot csak komoly ráfordítások árán lehet elvégezni.

Az alkalmazói szoftver

A személyi számítógépek felhasználója általában az alkalmazói programokkal kerül közvetlen kapcsolatba. Ennek megfelelően az alkalmazói programok sokrétűségén múlik, hogy a személyi számítógépek milyen hatékonysággal elégítik ki a felhasználók igényeit. A fejlett nyugati országok tapasztalatai alapján elmondható, hogy a személyi számítógép tulajdonosa a berendezés megvásárlását követő néhány éven belül többet költ szoftverre, mint amennyibe a hardver került.

A kívánalmaknak megfelelő szoftvertermék megszerzésére két lehetőség kínálkozik: vagy kész programokat vásárolunk, amennyiben kapható a piacon olyan termék, amelyet a kérdéses probléma megoldására készítettek, vagy a feladat pontos megfogalmazása után mi magunk készítjük el, esetleg megbízás alapján szakemberekkel készíttetjük el a programot.

Az egyedi, kizárólag a felhasználó igényeire szabott szoftvertermék előállítása igen költséges, mivel a megrendelőnek a program teljes kifejlesztésének költségeit meg kell fizetnie. A piacon kapható szoftvertermékek kevésbé speciálisak; eleve úgy tervezik őket, hogy a felhasználók minél szélesebb rétegének igényeit kielégítsék. Az eladott darabszámnak megfelelően a programok fejlesztési költsége is többfelé oszlik. Így fordulhat elő az, hogy az egyedi igényekre szabott szoftver ára 1–3 nagyságrenddel is meghaladhatja a piacon kapható „konfekcionált" programok árát.

Az eddigiekből következik, hogy amennyiben a felhasználó nem akar jelentős összegeket kifizetni a szükséges alkalmazói programok kifejlesztésére, a személyi számítógép vásárlásakor alapvető szempontként kell figyelembe vennie a kapható gépek szoftverellátottságát.

FINTA GYÖRGY

Iskolaszámítógép–BASIC–tanárok

Az iskolaszámítógép-program kezdő lépéseként Hajdú-Bihar megyében 1983 április–májusában a megye minden középiskolájából két-két tanár részt vett egy háromnapos, ún. „gépátadási" tanfolyamon. Itt minimális számítástechnikai és gépkezelési ismeretet (többek között a HT–1080Z BASIC nyelvének alapjait) sajátították el a résztvevők. A tanárok többsége vagy minimális számítástechnikai előismerettel rendelkezett, vagy azzal sem. Kevesen hallottak a mikroszámítógépekről, és nagyon kevesen dolgoztak már ilyen géppel.

A Megyei Pedagógus Továbbképzési Intézet decemberben hétnapos haladó, elsősorban a középiskolai számítástechnikai szakköri munkát elősegítő tanfolyamot szervezett a „gépátadási" tanfolyamon részt vett tanárok számára.

Előadóként vettem részt mindkét tanfolyamon. A Communications ACM folyóirat egyik cikke adta az ötletet, hogy a decemberi tanfolyam megkezdésekor minitesztet készítsek a BASIC programnyelv legalapvetőbb utasításainak működéséről alkotott elképzelésekről. A tanfolyam tagjai közül huszonkilencen vettek részt ebben a felmérésben.

A tanárok feladata az volt, hogy elemezzék, hogyan működik a megadott 8 utasítás és mi történik hatásukra.

A válaszokat 4 osztályba soroltam. *Helyes* – ha a válasz megfelelő volt, a kulcsszavakat tartalmazta és nem volt benne inkorrekt elem; *Nem teljes* – ha a válasz megfelelő, de hiányos volt, a kulcsszavakat csak részben tartalmazta és nem volt benne inkorrekt elem; *Hibás* – ha inkorrekt elem(ek)et tartalmazott; *Nincs válasz* – ha a kérdésre nem volt semmiféle válasz.

Például az IF A < B GOTO 99 utasítás esetén a helyes válasz: „Ha az A változó aktuális értéke kisebb, mint a B változóé, akkor a program a 99. sorszámú utasításon folytatódik. Ha A értéke nagyobb egyenlő a B értékénél, akkor a következő sor első utasítására tér rá a program." Nem teljes a válasz, ha csak az első mondat szerepel. Hibás viszont a „Hibajelzés történik" válasz.

Az utasítások és a válaszok megoszlása a helyes válaszok száma szerint növekvő sorrendben a következő:

Utasítás	Válaszok			
	Helyes	Nem teljes	Hibás	Nincs válasz
INPUT A	1	23	5	0
READ A	5	8	12	4
DATA 80, 90, 99	8	12	6	3
IF A < B GOTO 99	8	19	2	0
PRINT C	22	0	5	2
PRINT „C"	25	0	4	0
A = B + 1	25	3	0	1
60 GOTO 30	29	0	0	0

A továbbiakban közlöm az utasításoknál a helyes válaszokat (ezeket * jelöli), a leggyakoribb vagy legmeglepőbb hibás válasz(oka)t, a válaszolók számát, továbbá megjegyzéseket fűzök hozzájuk.

1. INPUT A

(*) A képernyőn megjelenik egy ? karakter 8
(*) A billentyűzetről be kell adni egy számot, amelyet NEW LINE zár le 3
(*) A számot elhelyezi az A változóba 3
Kiírja képernyőre az A-t 1
Értékadó utasítás 1

A 23 nem teljes válasz egy része akár hibásnak is minősíthető lenne, mert például az „A gép bekéri A-t" önmagában igaz, de az utasítás lényegéről nagyon keveset árul el. Az, hogy csak egyetlen helyes válasz volt, részben az INPUT utasítás összetett jellegéből, részben abból következhet, hogy alkalmazni könnyebb egy egy utasítást, mint programozási precizitással megmondani a pontos hatását.

2. READ A

(*) Az A változóban elhelyezi a DATA-lista következő elemét 5
DATA-listában elhelyezett adathoz hozzárendeli A-t 1
A program változója A 1
Beolvassa A-t 8

A legtöbb hibás választ tartalmazó utasítás. A „Beolvassa A-t" lakonikus választ hibásnak minősítettem, mert ez egyszerűen tükörfordítása az alapszónak, másrészt a beolvasás általában kívülről történő adatmegadást jelent. Itt van egyébként a legtöbb nincs válasz eset is.

3. DATA 80, 90, 99

(*) Olyan adatok tárolása a programban, amelyek READ utasítással érhetők el 8
Adatbeolvasás 1
A 80, 90, 99 sorszámú sorokban szereplő adatok 1
Adatátvitel a megadott számú sorokban 1

Az egyik legheterogénebb válaszegyüttes. Igen szélsőséges elképzelések is akadnak.

4. IF A < B GOTO 99

(*) Ha az A változó aktuális értéke kisebb, mint a B változóé, akkor a program a 99-es sorszámú utasításon folytatódik 19
(*) Ha A értéke nagyobb egyenlő a B értékénél, akkor a következő sor első utasítására tér rá a program 8
Hibajelzés történik 2

A válasz második mondata a válaszolók többsége számára kézenfekvő, ezért azt elhagyták, ezáltal viszont a válasz nem teljes. A két hibás „Hibajelzés történik" válasz esetén a THEN hiányát kifogásolták.

5. PRINT C

(*) Kiírja a C változó aktuális értékét 22
Kiírja C-t 4
Beolvas valamilyen adatot 1

A „Kiírja C-t" választ azért minősítettem hibásnak, mert a PRINT „C" utasításnál ugyanez a szöveg, vagy ennek változata szerepelt. A helyes választ adók kb. fele a válaszhoz hozzátette, hogy a kiírás képernyőre történik. Szintén kb. a fele a kiírja szó helyett a kinyomtatja szót használta. Egyetlen válaszoló sem jelölte meg a helyes képernyőpozícióit, ahová a kiírás történik, és próbálkozni is csak ketten mertek.

6. PRINT „C"

(*) Kiírja képernyőre a „C" betűt 23
C kiírása 2
Kiírja C értékét és a „C" betűt 2
Kiírja C-t, szóközöket hagyva 1
Beolvas valamilyen szöveget 1

Az előző utasításhoz fűzött megjegyzések ide is vonatkoznak.

7. A = B + 1

(*) Az A változó értéke legyen 1-gyel nagyobb, mint a B változó értéke 25

Itt nem volt hibás válasz. Igen sokan írták még pluszban az „Értékadó utasítás"-t is.

8. 60 GOTO 30

(*) A program a 30-as sorszámú utasításon folytatódik 29

Az egyetlen „telitalálatos" utasítás. A válaszolók fele hozzátette a „Feltétel nélküli ugrás" kiegészítést. Egyvalaki utalt arra, hogy 30-as sorszámú utasításnak léteznie kell a programban.

A megadott válaszsorozatok között nem volt olyan, amely csupa helyes választ tartalmazott volna, és olyan sem, amelyben ne lett volna legalább egy helyes válasz.

Amikor a felmérés készült, a tanároknak fél éve állt rendelkezésükre a számítógép. Az egész felmérés nem tudományos igénnyel, mély kapcsolatok feltárása céljából készült; inkább csak újabb színfoltot, érdekességet jelent az iskolaszámítógép-programmal kapcsolatban. A magam részéről a későbbi tanfolyamok számára kaptam némi információt arról, hogy mely területekre kell jobban odafigyelni.

Messzemenő következtetéseket levonni azonban nem lehet és nem is szabad. Mindenesetre elgondolkodtató az a tény, hogy a hibás és a nincs válasz esetek az összes válasz közel egyötödét jelentették. Igaz viszont, hogy ezeknek több mint fele a DATA és READ (egymással szoros kapcsolatban lévő) utasításokra esett. Ez az a két utasítás, amelyek alkalmazása általában nehézséget jelent (pontosabban: nem használják őket), jóllehet például játékprogram szinte elképzelhetetlen nélkülük.

Számomra a legérdekesebb válaszok a PRINT utasítással kapcsolatban érkeztek. Mint már utaltam rá, csak a helyesen válaszolók fele tartotta szükségesnek megjegyezni, hogy a kiírás a képernyőre történik, a válaszok másik fele a kinyomtatja szót tartalmazta. Ez nem véletlen. Későbbi tapasztalataim is azt erősítik meg, hogy a tanárok többsége számára a kiírás (nyomtatás) egyértelműen képernyőn való megjelenítést jelent. Kazettán általában csak a programot tárolják. A nyomtató pedig a pillanatnyi kiépítettséggel teljesen adekvát módon csak néhányuk képzeletében létezik – vágyálomként.

DR. JUHÁSZ ISTVÁN
KLTE Számológközpont, Debrecen

A SZEMÉLYI SZÁMÍTÓGÉPEK HATÁSA A CSALÁDRA, ISKOLÁRA AZ USA-BAN

Az amerikai családok egyre növekvő hányada vásárol és használ mikroszámítógépet. A következő években ezek a készülékek olyan általánossá válnak majd a háztartásokban, mint a mosógép vagy a kenyérszeletelő. Egyes családokban a számítógépet szórakozásra, máshol tanulásra vagy üzleti célokra használják. Az eltérő használati módok ellenére egy dologban megegyeznek a vélemények: a személyi számítógépek egyre növekvő szükséggel kerülnek be a családi életbe. A felnőttek és gyerekek több időt és pénzt költenek ezekre az eszközökre, mint azt valaki akár két évvel ezelőtt megjósolhatta volna. Tény, hogy a mikroszámítógépek nagyobb változást hoztak és hoznak az emberek életstílusában, mint a televízió, rádió vagy a mozi. Várható ez azért is, mert a számítógépek iskolákban történő terjesztése miatt a felnövekvő nemzedék számára használatuk teljesen természetessé válik.

Az USA-ban több, mint tíz éve indult egy központilag szervezett országos iskolaszámítógép-program, amely nemcsak a mikroszámítógépek iskolákban történő elterjesztéséről, hanem a hozzájuk tartozó oktatóanyagok kidolgoztatásáról is gondoskodik.

A program tapasztalatairól és általánosan a mikroszámítógépek elterjedésének társadalmi hatásáról tartott előadást 1984. május 10-én a Tudományszervezési és Informatikai Intézetben dr. Christopher Dede, a Houstoni Egyetem professzora. Előadása végén válaszolt az érdeklődő hallgatóság kérdéseire. Az alábbiakban röviden összefoglaljuk az elhangzottakat.

Az oktatás területén a személyi számítógépek használata szerint öt terület különböztethető meg:

- számítógéppel segített oktatás
- irányítótevékenység
- szervezés-vezetés
- információfeldolgozás
- programozási gyakorlat

A számítógép legnagyobb előnye, hogy a megsokszorozódott információmennyiséget nem kell fejben tartani (memória), ezért elsődleges feladat a szöveges információknak minél nagyobb mértékű számítógépre vitele. Így mód van arra, hogy a felhasználó ezekhez az információkhoz bármikor másodpercek alatt hozzájusson, azokat viszszakérdezhesse, sőt több területről kaphat egyszerre a munkájához szükséges adatokat, és a rendelkezésére álló gép számítási kapacitása szerint pl. különféle vizsgálatokat végezhet. Ehhez nem az információk ismeretére van szükség, hanem az információk rendszerezéséhez való képességére. Ennek fejlesztése az eligazodási készséget növeli, tehát nevelő hatása van.

A számítógépes oktatás eredményét vizsgálva arra a következtetésre jutottak, hogy két csoportba kell osztani az oktatásban résztvevőket: egyikbe azokat, akik majd mint számítástechnikai szakemberek fognak dolgozni, és a másikba azokat, akik az alkalmazók lesznek. Az ember és a gép együttműködésén alapuló programok korai időszakban is jelentkeztek az oktatás területén. Ezek a jövőben is a legfontosabb alkalmazások közé tartoznak majd. Az USA-ban már a 60-as évek során megfogalmazódtak azok a követelmények, amelyek a számítógépeknek az oktatás szolgálatába való állításához szükségesek.

Néhány éve komoly számítógépes programozás folyik már a különböző iskolákban. A tanítási folyamatban való alkalmazásnál újabban egyetlen dologra korlátozódott a felhasználási lehetőség, hogy a gép helyettesítse a tanárt. A gép a tanárokat azonban – magasabb rendű tanulási helyzetekben – nem pótolja, ezért a gépet azon a területen kell használni, ahol a tanár személyes jelenléte nélkülözhető, ill. az a tanár segítőeszközeként funkcionál.

Mivel egyetlen tanárra egy egész osztály jut, így nem biztos, hogy minden egyes diáknál sikerül a hiányokat felfedeznie és gyakoroltatnia azok kiküszöbölését.

A gépnek tehát legnagyobb előnye, hogy egyéni foglalkozást tesz lehetővé, sokszor elvégezteti ugyanazt a feladatot: a válaszokat gondosan értékeli, egyéni munkatempó szerint tanít, nem téved, a gép és diák kapcsolata személytelen, azaz a diák tévedésének nincs tanúja. Óvakodni kell azonban attól, hogy ne bízzuk mechanikusan a számítógépre az oktatást!

A tanárok fokozatosan építik be a tananyagba az iskolagépek használatát. Problémamegoldásra és gyakorlásra használják.

Az USA-ban sok tanulmány foglalkozik azzal, hogy milyenek az iskolai típusú relációk, milyen kapcsolat van a gép–diák, gép–tanár, és a gép használatán keresztül a tanár–diák, ill. diák–diák között.

Égetően szükséges, hogy a számítógépekhez a közeljövőben minél többen hozzáférjenek. A társadalom a problémák megoldásában csak akkor számíthat a számítógépek segítségére, ha sikerül egy számítógépen felnevelkedett generációnak felnőnie.

Törekedni kell arra, hogy a különböző társadalmi rétegek közötti számítógép-elosztás egyforma legyen. Tehát a szegényebb területek iskoláinak központilag több gépet kell juttatni, mert feltételezhető, hogy ezen iskolák diákjainak családjainál jóval kevesebb házi számítógép van, mint egyéb családoknál.

A mikroszámítógépek használata az amerikai családokban hosszú távon alapvető változásokat hozhat a kapcsolattartási formákban, beleértve ebbe a tanuláshoz való kapcsolatot is.

Minden valószínűség szerint a mikroszámítógépeknek személyiségformáló hatása is lesz azokra, akik napi idejükből elég sokat töltenek a gép mellett. Azok gondolkodásmódja, akik közvetlen kapcsolatban állnak számítógépekkel általában komplexebbé, rugalmasabbá lesz. Az előadó 1982-ben tanulmányt készített a „Személyi számítógépek szociális szerepe; beépülés a család szellemi életébe" címmel, és száz kiválasztott család alapján vont le tanulságokat. A megvizsgált családok 59%-ában egy családtag használta a gépet, 15%-ában több családtag egyénileg, 26%-ában csoportosan.

Az egyéni felhasználók a személyi számítógépet a következő célokra használták:

szórakozás	67%
tanulás	52%
munka	33%
kommunikáció	4%
programozás	52%
háztartás	19%

A felhasználók átlagosan napi fél–másfél órát töltenek számítógép mellett. Az igény ennél több lenne.

Milyen szabadidős tevékenységtől veszi el az időt a számítógép? A kísérletben résztvevőket megkérdezték, hogy milyen tevékenységet folytatnának „számítógépezés" helyett, ha az nem lenne lehetséges. 93%-uk a televíziózást választotta, 56% olvasást, 48% alvást. Mivel ezek mindegyike individuális tevékenység, úgy tűnik, hogy a számítógép a társas kapcsolatok rovására nem megy. Más oldalról viszont, sok esetben időmegtakarítást jelenthet a számítógép ésszerű használata, és ezzel további szabad idő keletkezhet.

Prof. Dede vizsgálatot folytatott a mikrogépek túlzott elterjedésének veszélyeiről és arról, hogy milyen hatással van a családokra, ha a gyermek túl sok időt tölt a számítógépek előtt. A túlzott egyoldalú időtöltés más műszaki eszközöknél is veszélyeket rejt magában. Ha pl. a tv-nézés veszélyeivel hasonlítjuk össze, akkor a mikrogépek javára billen a mérleg. Hiszen, amíg a tv-nézés passzív időtöltés, addig a mikrogépek használata aktív tevékenységnek minősül. A hazai számítástechnikai ismeretterjesztés megszabja a gondolkodásmódot.

A vizsgálatok alapján kiderült, hogy a számítógéppel rendelkező családokban általában egy személy foglalkozik komolyabban – naponta hosszabb időt rászánva – a géppel.

A társadalmi hatások között mindenképpen előnynek számít ezen területen a generációk, nemek közötti elmosódás.

A feltett kérdésekre adott válaszokból kiderült, hogy az USA-ban a mikrogépek elosztása arányaiban a következő:

általános iskola	30 gyerek/mikrogép
középiskola	20 gyerek/mikrogép

A szoftvert a diákok, tanárok ötletszerűen készítik és terjesztik. Szervezetten készítenek programcsomagokat megrendelésre egyes vállalatok, kb. 10 éve központilag is fejlesztenek iskolai szoftvereket.

Az iskolákban a gépek kiválasztása és elhelyezésük aránya

APPLE	49,4%
Radio Shack TRS–80 I, II	21,0%
Commodore	15,2%

A BASIC, PASCAL programozási nyelveket használják az általános iskola felső tagozatán és a középiskolákban, míg az alsó tagozaton a LOGO nyelvet, mely jó alapot biztosít a matematika megismeréséhez.

Arra a kérdésre, mit tesznek azért, hogy a programok hordozhatóak legyenek, azt a választ adta az előadó, hogy nagyobb cégek általában más nagyobb cégek gépeit emulálják a saját gépeiken.

Miután a számítógép oktatása az algoritmikus gondolkodást fejleszti, az egyik hozzászóló aggodalmát fejezte ki, nem tartanak-e attól, hogy ez az ugyancsak fontos intuíciós készség rovására megy. Dede professzor szerint ez valós probléma, tehát törekedni kell olyan gyakorlatok megoldására, ahol az intuitív és a kreatív gondolkodás is szerephez jut (pl. rajzolás gépi segédlettel, animáció, sakk stb.).

SIEGLER VERA, MAROSVÁRI ERIKA

Számítástechnika az oktatásban III.

E cikksorozat harmadik részeként kizárólag a HT–1080Z iskolaszámítógéppel foglalkozunk, és igyekszünk megmutatni, hogy a középfokú fizikaoktatásban és részben a tanártovábbképzésben lényeges anyagrészek elmélyítését hogyan segítheti a BASIC nyelvű, sokoldalú személyi számítógép.

A Fővárosi Pedagógiai Intézetben szervezetten és nagy ütemben folyik a tanártovábbképzés, amelynek kapcsán az egyes iskolákban bázisok alakulnak ki – legalább egy tanár már jól képzett a számítástechnikában –, és így a rendelkezésre álló iskolaszámítógépet érdemben tudják alkalmazni. Több iskola és a Fővárosi Pedagógiai Intézet központi szervezésben is működtet különböző számítástechnikai szakköröket, ahol a tanulók sajátíthatják el e kétségtelenül újszerű, most már tudományágnak nevezhető tudásanyag részleteit.

Ehelyütt magára a BASIC nyelvre, annak különböző szintű ismeretére és szorosan ehhez kötve az egyes tantárgyakban való alkalmazására kell gondolnunk. Iskolai vonatkozásban főként a matematika, fizika, kémia tantárgy területeiről van szó. Hosszabb időn keresztül az egész országban a középfokú oktatásban a HT–1080Z számítógép képezi majd alapját mindenfajta számítástechnikai tevékenységnek. Ezt kell megismerni és lehetőségeit kiaknázni, amennyire csak lehet.

E terület kifejlesztését megkönnyíti, hogy a fiatalok – már akikben megvan a szellemi fogékonyság és a magasabb rendű iránti érdeklődés – meglepő gyorsasággal, éles felfogással és kedvvel sajátítják el a számítástechnikai ismereteket, főképpen ha értelmét látják, azaz, ha nem öncélú ismeretként kell megtanulniuk. Ettől szerencsére távol állnak a számítástechnika elterjesztésében közreműködő tanárok és képzett tanulók törekvései, mert a számítástechnikát szerves egységben tanulják, alkalmazásával együtt.

Jellemző még az ilyen irányú képzésre, hogy diákszakköröken sokkal jobban elterjedt az önállóság, mint más szakköröknben. Egy idő után a tehetséges tanulók a saját útjukra térnek, sőt más, hasonló célú szakörök vezetésével is megbízhatók.

Ilyen, optimálisnak nevezhető esetekben a tanári beavatkozás pusztán konzultációszerű. Az eddigi tapasztalatok szerint a számítástechnikai kultúra fiatalos gyorsasággal és eredményesen terjed.

A következőkben olyan problémákkal foglalkozunk, amelyek részletes elemzése tanárok számára számítástechnikai továbbképzésnek is tekinthető.

Az egyes problémák tárgyalásakor felhívjuk a figyelmet a kilistázott programokban szereplő érdekességekre, amelyek valójában a gép sokoldalúságát mutatják. Ilyen például az IF – THEN – ELSE vagy a FOR – TO – NEXT utasításokban rejlő, igen nagyfokú változatosság.

Pontszerű töltések erővonalképe

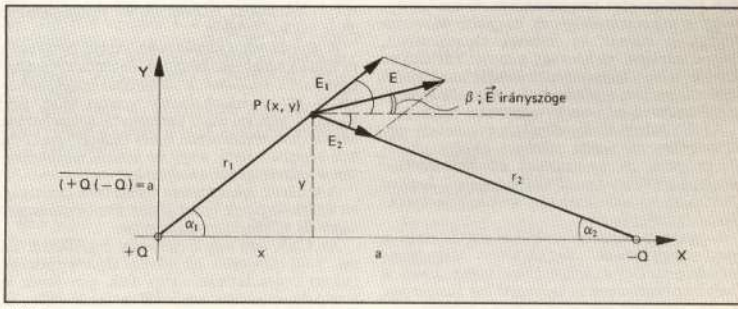
Tekintsünk két egyenlő nagyságú, ellentétes előjelű pontszerű elektromos töltést. A szokványos iskolai kísérletben ezek erővonalképét van de Graaf-féle generátorral, glicerinbe helyezett búzadaraszemekkel, köztük két feltöltött kis alumínium tárcsával mutatjuk be.

Most viszont meghatározzuk az erővonalaknak, mint görbeseregnek az iránymezőjét, és szukcesszív approximációval, az y' iránymező x szakaszonkénti linearizálásával állítjuk elő a görbesereget, majd ennek készítjük el a programját.

Az oktatásban ez két lépcsőben használható. Tekinthetjük a matematikai elemzést és a vonatkozó programot egyszerűen eszköznek, és csak magát a grafikus képet mutatjuk be. Második lépésként a matematikai hátteret is tárgyalhatjuk, ha az y' deriváltat differenciahányadosként tekintjük közelítéssel, és ekkor önként adódik a program megértése is.

Végül ebből y'-re kapjuk, hogy

$$y' = \frac{y}{x + \frac{a}{\left[\frac{(a-x)^2+y^2}{x^2+y^2}\right]^{3/2} - 1}} \qquad (2)$$



1. ábra

A matematikai rész

Az alapgondolat az, hogy az $\vec{E}$ térerősség vektormező a keresett görbesereg iránymezője, tehát a görbesereg egyik egyedének P pontbeli y' érintő iránytangense a P-beli $\vec{E}$ vektor irányszögének tangense. Ez viszont az $\vec{E}$ vektor Y és X irányú komponense hányadosaként adódik. Maga az $\vec{E}$ vektor két komponensből tevődik össze: külön a Q és külön a (–Q) töltés térerősségéből *(1. ábra)*.

$$\operatorname{tg}\beta = \frac{E_y}{E_x}; \qquad y'(x,y) = \operatorname{tg}\beta$$

$$E_y = E_{1y} + E_{2y}; \qquad E_y = E_1 \sin\alpha_1 - E_2 \sin\alpha_2$$

$$E_x = E_{1x} + E_{2x}; \qquad E_x = E_1 \cos\alpha_1 + E_2 \cos\alpha_2$$

$$E_1 = k\frac{Q}{r_1^2}; \qquad E_2 = k\frac{Q}{r_2^2}. \qquad (1)$$

A $\operatorname{tg}\beta = y'(x,y)$ egyenlőség fejezi ki, hogy az $\vec{E}$ iránymező a görbesereg érintőinek seregét adja, így y'-vel folytatva a számítást, az ábra szerint

$$y'(x,y) = \frac{E_1\frac{y}{r_1} - E_2\frac{y}{r_2}}{E_1\frac{x}{r_1} + E_2\frac{a-x}{r_2}}.$$

Ide behelyettesítjük a (1) coulomb erőt:

$$y'(x,y) = \frac{\frac{y}{r_1^3} - \frac{y}{r_2^3}}{\frac{x}{r_1^3} + \frac{a-x}{r_2^3}}.$$

A program matematikai előkészítése

Az *a* szakaszt lehetőleg kicsi Δx szakaszokra osztjuk. Ezeken belül y' függvényt állandónak tekintjük. Az új y'-t mindig az előzőekre építve, rekurziós egyenletrendszerrel határozzuk meg. Így tetszőleges x értéket és a hozzá tartozó y ordinátát megadhatjuk, ha figyelembe vesszük y'-nek differenciahányadosként való felírását: $y' \sim \Delta y/\Delta x$. A továbbiakban egyenlőségjelet használunk. Tehát:

$$x = n\Delta x$$

$$\Delta y = y'(x,y)\Delta x$$

$$y = \sum_{k=1}^{n} \Delta y_k$$

ahol *n* futóindex. Ezek alapján az ábrázolásra kerülő (x, y) pontpárokra felírhatjuk a rekurziós egyenleteket, amelyek a szukcesszív előrehaladást tartalmazzák:

Kezdő x_1 és y_1 legyen az önkényesen megadott Δx és Δy. Elegendő kicsire csak Δx-et kell választani, Δy akármekkora lehet, mivel y' linearizálása Δx által a Δx intervallumban valósul meg. Az $x_1 = \Delta x$ és $y_1 = \Delta y_1$ pontpárral az önkényesen választott első görbére álltunk rá, a következő Δy csak Δx-ben való teljes végigfutáskor következik, miáltal az önkényesen választott második görbére álltunk. Így haladva kapjuk, hogy

$$x_1 = \Delta x; \qquad y_1 = \Delta y$$

$$x_2 = 2\,\Delta x;$$

$$\Delta y_2 = y'(x_1, y_1)\cdot\Delta x$$

$$y_2 = \Delta y_1 + \Delta y_2$$

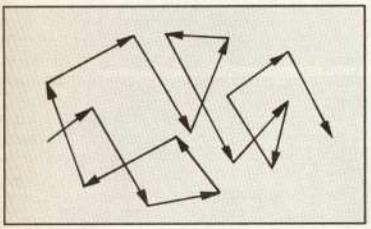
$$x_3 = 3\,\Delta x$$

$$\Delta y_3 = y'(x_2, y_2)\cdot\Delta x$$

$$y_3 = \Delta y_1 + \Delta y_2 + \Delta y_3.$$

```
10 PRINT@975,"+Q,-Q EROVONALKEP";
20 READ A,B,L
30 DATA 1,1,30
40 FOR M=1 TO 5
50 Y=M*B
60 FOR N=1 TO L
70 X=N*A
80 SET(10+X,23-Y)
90 SET(10+X,23+Y)
100 C=((L-X)[2+Y[2)/(X[2+Y[2)
110 IF C[(3/2)-1=0 THEN 160
120 D=L/(C[(3/2)-1)
130 IF X+D=0 THEN 160
140 T=Y/(X+D)
150 Y=Y+T*A
160 NEXT N
170 NEXT M
180 GOTO180
```

1. program



2. ábra

Végül a programozható rekurziós egyenletrendszer az alábbi, ahol y'(x, y) függvény konkrét alakja a (2) képlettel adott:

$$x_{n+1} = (n+1)\Delta x$$

$$\Delta y_{n+1} = y'(x_n, y_n) \cdot \Delta x$$

$$y_{n+1} = \sum_{k=1}^{n+1} \Delta y_k.$$

A program értelmezéséhez

A 20. és 30. sorban a következő hozzárendelést végeztük:

A	x_1	1
B	y_1	1
L	a	30

Az 1, 1, 30 önkényes egységeket jelent. Ezekkel és a 80. sor SET utasításával az ábra a képernyő közepére kerül. Az 50. és 70. sor alapján A és B egyben a Δx és Δy közöket is jelenti. A 110. és 130. sorral kritikus x helyeket zárunk ki: ugyanis $y' = 0$ úgy valósul meg, hogy (2)-ben a nevező végtelenhez tart. Észrevehetjük, hogy két, egymásba ágyazott ciklussal hozzuk létre a görbesereget: amíg N le nem fut 30-ig, a kezdő Y érték nem változik, hanem erre építve számítjuk ki mindig az aktuális x-hez tartozó y értéket.

A program megfelelő átalakítással azonos előjelű, egyenlő nagyságú töltések erővonalképének megjelenítésére is alkalmas.

Ionok driftmozgása

Ismeretes, hogy az atomok, molekulák állandó, rendezetlen mozgást végeznek. Azzal, hogy adott hőmérsékletű a vizsgált anyag, részecskéinek ez a termikus mozgása mindig megvan. Ha például valamely elzárt gáz molekuláit vagy – nemesgáz esetén – atomjait ionizáljuk, úgy azok elektromosan töltöttek lesznek, és általában néhány elektron nagyságú töltéssel rendelkeznek. Szokott eset az egyszeres ionizáció. Ez elektromos térben ütközési ionizációval, vagy még termikus úton is létrehozható. Ha ionokat vizsgálunk, melyek homogén elektromos térben vannak, mozgásuk viszonylag könnyen leírható.

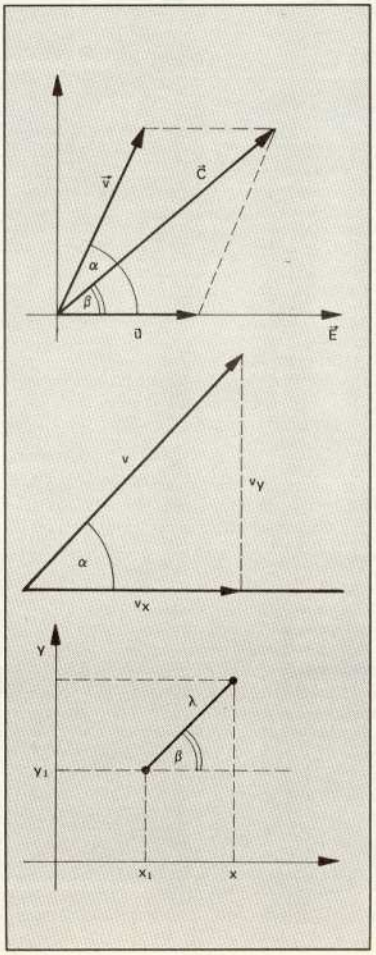
Ez esetben a termikus mozgásra szuperponálódik egy erőtér irányú mozgás. A kép ilyenkor az, hogy a zegzugos mozgást végző ion sodródik az erőtér irányában *(2. ábra)*.

Ezt a jelenséget szeretnénk most matematikailag megragadni és programba áttenni.

Legyen a termikus mozgás sebességátlagának nagysága (vagy négyzetes átlagsebessége) v, az $\vec{E}$ erőtér irányú átlagsebesség nagysága u. Minthogy az ionok átlagosan λ szakaszonként (szabad úthossz) ütköznek, ezek az ütközések egy $\vec{E}$ irányú átlagsebességet determinálnak. A λ szakaszon belül ütközés nélkül gyorsul $\vec{E}$ irányban, majd rugalmasan, centrálisan ütközik a vele egyenlő nagy tömegű másik ionnal. Ütközés után minden irány bizonyos valószínűséggel bír. Ez vonatkozik arra az esetre is, ha nincs sodrás, azaz driftmozgás, tehát például elektromos erőtér nélkül a gáz egy edényben van elzárva. Ekkor is folyamatosan rugalmas ütközéseket képzelhetünk el, így egy-egy ütközés után a v sebességvektor iránya (0, 360) fokos intervallumban oszlik el. Lényeges különbség az előbbihez képest, hogy most minden irány egyenlően valószínű. Az eredő c sebességre áll, hogy $\vec{c} = \vec{v} + \vec{u}$, amelyben már a driftmozgás is jelentkezik. A $\vec{c}$ sebesség statisztikus jellegét $\vec{v}$ adja, $\vec{u}$ az erőtér irányú részt, és így $\vec{c}$ irányeloszlása már nem egyenletes $\vec{u}$ miatt. Az eredő $\vec{c}$ szögeloszlásában valószínűbbek az $\vec{E}$ térerősségvektor irányába mutató szögek, mint az azzal ellentétes irányúak. Elképzeléseink valójában csak ideális gázra jók. De azért a jelenség lényegéről így is számot adhatunk.

A termikus eredetű $\vec{v}$ sebesség irányeloszlására a számítógép véletlenszám-generátorát használjuk egy RND (x) utasítással, majd képezzük az eredő $\vec{c}$ sebességet. Ennek irányszöge meghatározza ezt az egyenest, melyből λ hosszúságú szakaszt kell ábrázolnunk. Ehhez, ennek végpontjához csatlakoztatjuk a következő λ hosszúságú szakaszt, de már más irányítással, ahogy azt az új eredő c sebességvektor előírja.

3. ábra



2. program

```
10 PRINT"IONOK DRIFTMOZGASA"
20 READX1,Y1,U,V,L
30 DATA 20,23,3,5,3
40 RANDOM
50 DIM A(36)
60 DIM C(36) , B(36)
70 FOR I=1 TO 36
80 A(I)=RND(36)
90 A1=A(I)*10*0.0174533
100 REM A2=A(I)*10
110 C1=U+V*COS(A1)
120 C2=V*SIN(A1)
130 IF SGN(C1)=0 THEN70
140 TB=C2/C1
150 CB=1/SQR(1+TB[2)
160 C=SQR(C1[2+C2[2)
170 B1=ATN(C2/C1):B=ABS(B1)*
    180/3.141592
180 IF C1>0 THEN 200
190 IF C2>0 THEN 210 ELSE 230
200 IF C2>0 THEN 220 ELSE 240
210 B=180-B:GOTO 600
220 B=B    :GOTO 600
230 B=180+B:GOTO 600
240 B=360-B:GOTO 600
250 GOSUB 500
260 FOR X=X1 TO R STEP Q
270 Y=Y1+TB*(X-X1)
275  IF Y<=0 OR Y>47 THEN 350
280 SET(X,Y)
290 D=ABS(X-X1)/CB
300 IF D>=L THEN 320
310 NEXT X
320 X1=X
330 Y1=Y
340 NEXT I
350 STOP
360 REM A DIM SEB.ES IR.SZO-
370 REM GEKET KIIRATJUK
375 K=0 : M=0
380 FOR K=1 TO 36
390 PRINT"C=";USING"##.#";C(K);
395 PRINT TAB(8)"M/S;";
400 M=M+1
410 PRINT TAB(15)"IRANY;B=
    ";USING"###.#";B(M);
415 PRINT TAB(29)"FOK"
420 NEXT K
430 END
500 IF C1>0 THEN 520
510 R=-127:Q=-1:GOTO 530
520 R=127 :Q=1
530 RETURN
600 K=K+1
610 M=M+1
620 C(K)=C
630 B(M)=B
640 GOTO 250
```

A matematikai rész és a program előkészítése

Válasszuk az $\vec{E}$ térerősség irányát x iránynak. Ekkor az ábrák *(3. ábra)* alapján végezhetjük a számítást:

$$u_x = -u; \quad u_y = 0$$
$$v_x = v\cos\alpha; \quad v_y = v\sin\alpha$$
$$c_x = u + v_x; \quad c_y = v_y$$
$$\operatorname{tg}\beta = \frac{v\sin\alpha}{u + v\cos\alpha}$$

A β eredő irányszöget (két ütközés között) az (1) és (2) egyenletek adják. Az egyes β irányú egyenesekből λ hosszúságú szakaszok ábrázolását és folytonos egymásutáni összeillesztését a β irányú, egy ponton átmenő egyenes egyenlete adja, és egy λ-ra kirótt egyenlőtlenség, amely az alábbi:

$$\cos\beta = \frac{x - x_n}{d}; \quad d = \frac{x - x_n}{\cos\beta}; \quad (3)$$
$$d \le \lambda,$$

ahol az egyenesből $d \le \lambda$ szakaszon ábrázoljuk az egyes pontokat SET (x, y) utasítással úgy, hogy a λ szakasz végén lévő P (x, y) pont koordinátái lesznek az új λ szakasz x_1, y_1 kezdőpont koordinátái. Az (1), (2) és (3) formulák egyben a *2. program* alapját képező formulák is.

A program 2Ø. és 3Ø. sorában az alábbi hozzárendelések érvényesek:

XI	x_1	20
YI	y_1	23
U	u	3
V	v	5
L		3,5

Az u, v értékek célszerűen másnak választhatók. Érdekes az u = 0, v ≠ 0 és a v = 0, u ≠ 0 eset. Az első a tiszta termikus mozgást adja, drift nélkül, ami fizikailag is realizálható, míg a második a tiszta driftmozgás, ami fizikailag csak határeset, mivel termikus mozgás minden hőmérsékleten van.

A programban megjelenítjük a driftmozgást a zegzugos szakaszok eltolódásával és külön – két DIM utasítással – kiíratjuk az egyes *c* sebességeket és azok irányszögét.

A változókat úgy választottuk, hogy áttekinthető legyen a program: az egyes mennyiségek matematikai jelöléseire emlékeztetnek. Például TB jelenti tg β-t vagy B a β szög fő értékét.

Lényeges része a programnak, hogy a mindig aktuális iránnyal kell megadnunk a sorok által megjelenített egyenes szakaszok helyzetét. A helyes irányok kiválasztása – vagyis a *c* sebesség szögének meghatározása – a 18Ø–24Ø. sorokban történik, hiszen 17Ø-ben B csak a β szög fő értékét szolgáltatja.

Minden egyenes szakaszhoz tartozik egy |c| sebesség és egy irányszög. Ezeket az összetartozó párokat a szakasz kirajzoltatásával egyidőben elraktározzuk két, 36 elemű sorvektor dimenzionálásával. Miután a grafikus megjelenítés véget ért, egy CONT utasítással történik az adatpárok kiíratása.

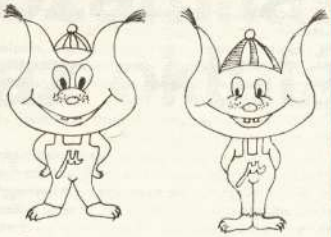
Igen jól érzékelhető a drift, ha u > 0; v > 0, és jól jelentkezik a tiszta termikus mozgás, ha u = 0; v > = 0. Több esetben a λ szabad úthossz-szakaszok elérik a képernyő felső vagy alsó szélét, ekkor a gép FC hibát jelez.

Hogy ez ne történjék meg, beiktattuk a 275. sort. Ilyen esetben három lehetőség van: fizikailag a leginkább reális, ha egyszerűen leállítjuk a jelenséget és ráállunk a 35Ø. sorra. Ez esetben nem kapott értéket valamennyi tömbelem.

Az egész jelenséget úgy tehetjük szemléletessé, ha többször állítjuk elő elejétől indítva. A véletlen eloszlás következtében mindig más és más lesz a driftmozgás.

DR. WIEDEMANN LÁSZLÓ
Fővárosi Pedagógiai Intézet

Kabalapályázatunkra érkezett



1. *Stark Gusztáv, Budapest*

Iskolaszámítógép-pályázat és sakkoktatás

A Neumann János Számítógéptudományi Társaság újonnan alakuló Sakkprogramozási Munkabizottsága, valamint a Magyar Sakk Szövetség Számítógép Bizottsága felkérésére a „Középiskolai számítógépes oktatási programpályázat" meghirdetője, a Tudományszervezési és Informatikai Intézet ezúton is megerősíti, hogy a pályázat keretében például a középiskolák sakk szakkörei részére hasznosítható sakkoktató programot pályázatként be lehet adni.

Örömmel veszi a fent említett bizottságok felajánlását a módszertani bírálatok elvégzésére, és erre való tekintettel közli az általuk megkívánt feltételeket, amelyek természetesen nem kizárólagosak:

A cél olyan program vagy programcsomag készítése, amely HT–1080Z típusú számítógépen a sakk alapfokú ismereteinek elsajátítását teszi lehetővé. Magában kell foglalnia a sakkszabályok, a sakkjáték legfőbb alapelvei, a leggyakrabban alkalmazott megnyitások és az alapvető végjátékok ismertetését.

Az említett bizottságok szakértői a részt venni szándékozóknak – a szerkesztőséghez intézett levélbeni kérdések alapján – szívesen szolgálnak szaktanáccsal. Az oktatóprogram önmaga nem kell, hogy sakkozni tudjon (ez meghaladná az iskolaszámítógépek lehetőségeit), de – dokumentációs jelleggel – mintajátszmákat, illetve játszmákból vett hadállásokat természetesen tartalmazhat.

(A lapunk ez évi 1. számában kiírt Sakkprogramozási pályázat az iskolaszámítógép-pályázattól független. – A szerk.)

VAN MÉG EGY DOBÁSA?

1981 szeptemberében az Országos Oktatástechnikai Központ, a Fővárosi Pedagógiai Intézet és az Országos Pedagógiai Intézet „Zsebszámológépek és személyi számítógépek az oktatásban" címmel kísérletet indított több általános, közép- és felsőfokú intézmény részvételével. A Nyíregyházi Tanárképző Főiskola Tanítóképző Intézete – tanító szakos hallgatók egy csoportjával – speciális kollégium keretében bekapcsolódott a kísérletbe. Az előző két évben a PTK–1072 és a PTK–1050-es programozható zsebszámológépek, az idén a HT–2080Z személyi számítógép oktatásban való alkalmazásának lehetőségeit vizsgáltuk, illetve vizsgáljuk.

E keretek között a HT–2080Z személyi számítógépnek egy konkrét szemináriumi alkalmazását ismertetjük. Azt, hogy hogyan alkalmaztunk egy egyszerű, kockadobást szimuláló programot az eseményalgebra, valószínűség értelmezése, teljes eseményrendszer anyagrészek tárgyalásakor.

A programot úgy készítettük el, hogy kiírta a képernyőre a dobások számát (N), a dobásértékhez tartozó egyes elemi események (1-es, 2-es, 3-as, 4-es, 5-ös, 6-os dobás) gyakoriságát, relatív gyakoriságát (R1, R2, R3, R4, R5, R6), a relatív gyakoriságok összegét, valamint grafikonon ábrázolta az egyes elemi eseményekhez tartozó relatív gyakoriság értékeket.

A programot a következőképpen alkalmaztuk a szemináriumon. Először egy kockadobást végzтettünk a számítógéppel: N = 1 *(1. kép)*.

Ez a dobás 5-ös volt. Így az 5-ös dobáshoz 1 gyakoriság, 1 relatív gyakoriság tartozott (R5 = 1), és a grafikon ennek megfelelően 5-nél 1-et mutatott.

5 kockadobás esetén *(2. kép)* még mindig igen nagy az eltérés az egyes elemi események relatív gyakoriságai között.

N = 60 esetén *(3. kép)* a relatív gyakorisági értékek egyre jobban közelítenek egymáshoz, amit a grafikon is érzékeltet.

100 dobás után *(4. kép)* a relatív gyakoriságok közötti maximális eltérés – ebben az esetben – már csak 0,11. Ez az eltérés N = 6000 esetén *(5. kép)* már kisebb, mint 0,014.

10 000 dobásnál *(6. kép)* az egyes elemi eseményekhez tartozó relatív gyakorisági értékek közötti maximális eltérés már csak 0,0052.

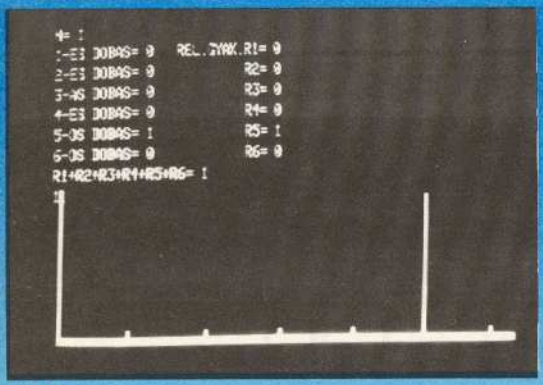
A hallgatók tapasztalták, hogy minél nagyobb a dobásszám, annál inkább közelítenek az elemi események relatív gyakoriságai egy meghatározott 1/6 = 0,1666... értékhez. (Ez a közelítés 10 000 dobás esetén már igen jól érzékelhető.)

Ezután tértünk rá az eemény valószínűségének definiálására. Azt a számot, amely körül a véletlen esemény relatív gyakorisága ingadozik, az esemény valószínűségének nevezzük.

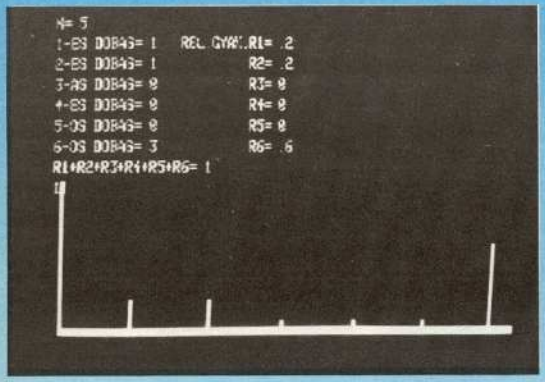
Az előző évekhez viszonyítva a személyi számítógép alkalmazása érthetőbbé tette, elősegítette ennek az anyagrésznek a megértését.

DR. ISZÁJ FERENC
főiskolai adjunktus
Bessenyei György Tanárképző Főiskola
Tanítóképző Intézet

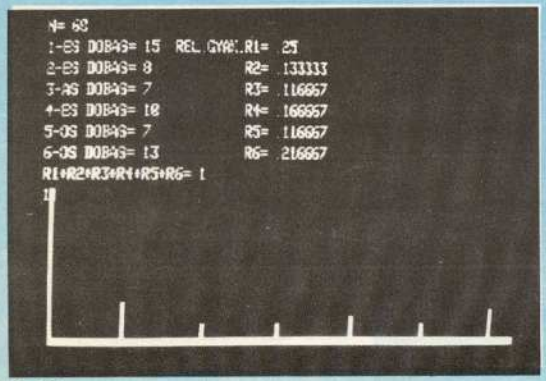
Valószínűségelmélet a gyakorlatban



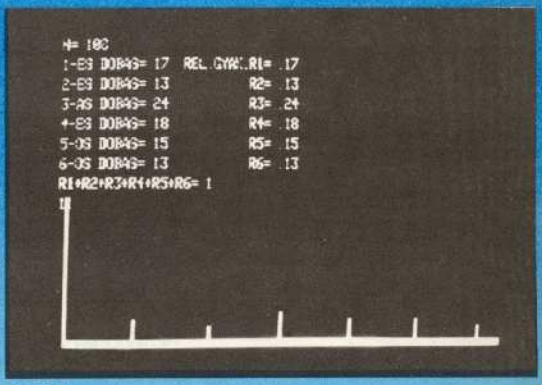
1. kép



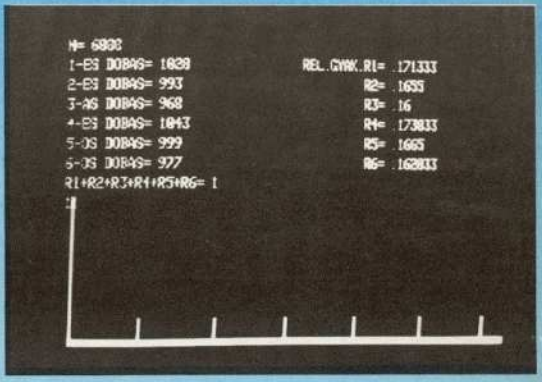
2. kép



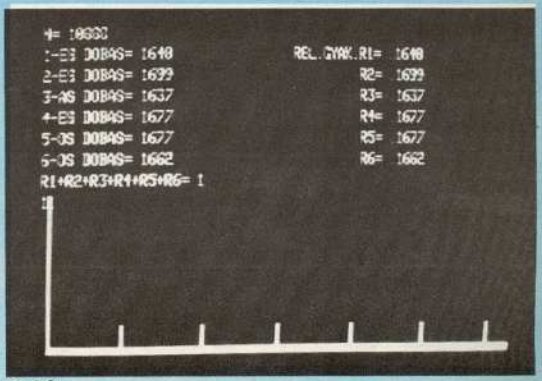
3. kép



4. kép



5. kép



6. kép

Paneljavítás hagyományos módszerrel.
Látható, hogy a hibakereséshez sok műszer szükséges, a bemenőjeleket kézzel állítják be és a jeleket oszcilloszkóppal és multiméterrel követik.
A hibakeresés esetenként több órát, jó és speciális gépismeretet, magas fokú szakmai felkészültséget követel.

Az irodagépek javítási technológiája

Az utóbbi időben forgalomba került gépek moduláris felépítésűek. Ez azt jelenti, hogy a javítás két fázisra osztható:

1) Helyszíni javítás

- kisebb mechanikai hibák
- beállítások
- elektronikus hibák javítása kártyacserével vagy egységcserével

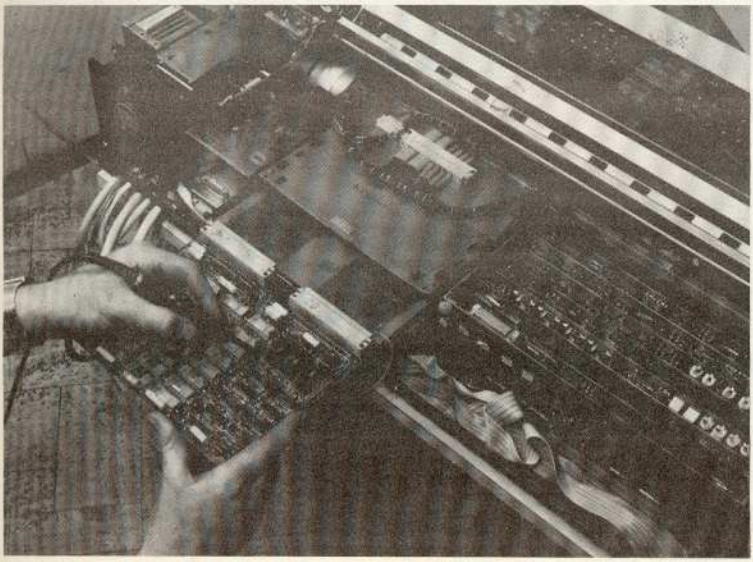
2) Műhelyjavítás

- a kártyák, csereegységek hibakeresése
- a hibás egységek javítása
- a megjavított egységek ellenőrzése, esetleg tartós tesztelése

Mivel a felhasználók igénye, hogy mielőbb újból üzemképes legyen a gépük, a csereegységgel történő javítási mód világszerte egyeduralkodó.

Mikroszámítógépek alkalmazása digitális áramköri kártyák javítására a szervizgyakorlatban

Az Információtechnikai Vállalat által ellátott elektronikus gépek választéka rendkívül széles. Komoly problémát jelent a szervizmunkához szükséges felszerelések (műszerek, csereegységek, háttérgépek és alkatrészek) biztosítása, esetenként a gyártóművek késedelmes szállításai miatt, de elsősorban a pénzügyi problémák miatt. (Egy-egy új géptípus bevezetése sokmillió forintos készletnövekedést és fejlesztési alap igényt jelent.)



Korszerű gép – hagyományos javítási mód.
robotron A 5110 típusú gép javítása.
Látható, hogy a gép felépítése bonyolult, sok alkatrészt tartalmaz, így a hibabehatárolás rendkívül munka- és időigényes.

Az SZKI által gyártott SKT vizsgálókészülék. A vizsgálókészüléken kívül az esetek többségében csak egy logikai szintjelző szükséges. A hiba behatárolása néhány perc alatt megtörténik.

A készülékek két fő részből állnak:
a) vezérlő számítógép
b) vizsgáló adapter

Vezérlő számítógépként minden olyan gép alkalmazható, amely hajlékonylemezzel és megfelelő operációs rendszerrel rendelkezik.
A vizsgáló adapter a számítógépből byte-onként érkező adatokat a be- illetve kimeneti pontokra szétosztja, valamint ezek állapotát visszajelzi.

A vizsgált kártyák csatlakozási pontjainak irányát (be- vagy kimenet) és a bemenetek jeleit a programban tároljuk. A program futása során a bemenetekre előírt sorrendben különböző jeleket adunk és vizsgáljuk a kimenetek állapotát. A szükséges válaszjelek szintén a programban vannak tárolva. Eltérés esetén a program futása leáll, és a hibás állapot a kártyán fennmarad.
Ekkor a hibás alkatrész egyszerű eszközökkel (pl. logikai szintjelzővel) megkereshető.

A számítógépes hibakeresés bevezetése és tapasztalatai

A számítógépes hibakeresés mielőbbi bevezetése érdekében a munkát az SZKI TS 51 intelligens terminálján alapuló SKT (statikus kártya Teszter) rendszerével kezdtük meg.
Elsőként a robotron 1370 gépcsaládnál vezettük be ezt a hibakeresési módszert. A tapasztalatok nagyon kedvezőek, a hibabehatárolás közel 100%-os, ami azt jelenti, hogy a tesztprogrammal jónak bizonyult kártyák üzemi körülmények között is hibátlanok.
Annak érdekében, hogy a szervizeinket minél nagyobb számban elláthassuk ilyen vizsgálókészülékekkel, saját olcsó mikroszámítógép és vizsgálóadapter kifejlesztését végezzük.

A hibás kártyák és csereegységek javítása

A hibás kártyák és csereegységek javítása a műhelyben hagyományosan háttérgépek segítségével történik. Ennek több hátránya is van.
A háttérgép ugyanolyan adatfeldolgozó gép, mint amilyennek a csereegységét javítani kívánjuk. Ezért a vele való hibakeresés továbbra is hosszadalmas és speciális gépismeretet igénylő munka. További probléma, hogy a különböző típusvariációk nem ritkán más gépet igényelnek.

Vagyis a háttérgéppel való javítás speciális felszerelést és ismereteket igényel, ugyanakkor termelékenysége – a típushibáktól eltekintve – rendkívül alacsony. Fenti hátrányok miatt kezdtünk el új javítási technológia kifejlesztésével foglalkozni.
Az új koncepció szerint célul tűztük ki univerzálisan felhasználható, vagyis több géptípusnál, illetve egységnél alkalmazható hibakereső berendezések kifejlesztését.
Már kezdetben egyértelmű volt, hogy a kitűzött célt mikroszámítógépek alkalmazásával érhetjük el. A kifejlesztendő eszközöket két csoportba osztottuk:

Típusorientált eszközök

Ezek egy vagy néhány géptípus javításánál alkalmazhatók tesztelésre, hibakeresésre. (Ilyenek pl. a különböző perifériajavítók.) A feladatuk viszonylag egyszerű, így kevés alkatrészből hordozható kivitelben is elkészíthetők.

Univerzális, programozható vizsgálókészülékek. Ezek feladata a csereegységeken a hibabehatárolás, illetve a javítás utáni ellenőrzés elvégzése. A műhelyjavításoknál ma már nélkülözhetetlenek az ilyen vizsgálókészülékek. Ismerkedjünk meg ezen készülékek felépítésével és működésével.

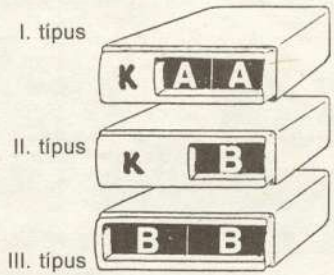
A robotron 1370 típusú gépe. Ennek kártyáit már több mint egy éve számítógépes hibakereséssel javítjuk (SKT rendszer). A javítás után a kártyák ellenőrzése ezen a háttérgépen történik.

A PROPER személyi számítógépek konstrukciós lehetőségei

A PROPER professzionális személyi számítógép-család egységes méretű készülékházzal rendelkezik. Ez lehetővé teszi, hogy a felhasználó a moduláris kialakítású különböző háttértárolókat is tartalmazó készülékek közül a számára optimálisat válassza.

A készülékházban a háttértárolók különböző méretű műszerfiókban („A", ill. „B" fiók) helyezkednek el.

Készülékház típusaink:



ahol az „A" műszerfiókban 1 db 5,25″ méretű,
a „B" műszerfiókban 1 db 8″ méretű
háttértároló található.

A készülékház típusok közül az I. és II. típus a háttértárak mellett a központi egységet (az ábrán K betűvel jelölve) is magában foglalhatja.

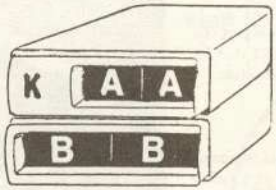
Milyen háttértárak alkalmazhatók?

A készülékház áramellátó egysége számos kiépítést tesz lehetővé.

Az „A" műszerfiókban vagylagosan
1 db 5,25″ méretű floppy diszk egység
1 db 5,25″ méretű merevlemezes diszk egység
2 db 5,25″ méretű, félmagasságú (ún. „slimline") floppy diszk egység,

míg a „B" műszerfiókban (ugyancsak vagylagosan)
1 db 8″ méretű floppy diszk egység
1 db 8″ méretű merevlemezes diszk egység helyezhető el.

Nos, ebből számtalan variációs lehetőség adódik, figyelembe véve, hogy a PROPER-család egyidejűleg 4 háttértárat kezelhet. Például egy széleskörűen használható floppy diszk alapú konfiguráció

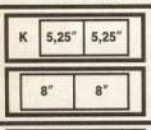
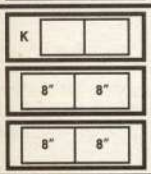
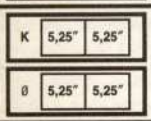
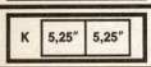
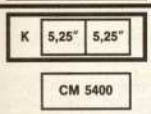
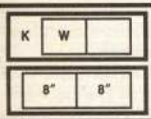
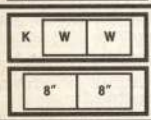
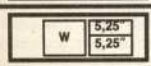


Háttértár kapacitása
megközelíti az 1,5 Mbájtot

Milyen kialakítású munkahelyek állnak rendelkezésre?

A személyi számítógépeinkből kialakított konfigurációkat (beleértve a képernyőt, a billentyűzetet és a nyomtatót is) kívánság esetén *munkaasztalba* építve is szállítjuk. A munkaasztalt ergonómiai szempontok figyelembevételével alakítottuk ki. Konstrukciója lehetővé teszi, hogy az esetleges további bővítési igényeknek is megfeleljen.

1015 Budapest
Donáti u. 35–45.
Tel.: 260-000

PROPER HÁTTÉRTÁR VÁLTOZATOK		PROPER-8	PROPER-16
	2 db 8"-os floppy diszk és 2 db 5,25"-os floppy diszk	különösen ajánlott	különösen ajánlott
	4 db 8"-os floppy diszk	ajánlott	ajánlott
	4 db 5,25"-os floppy diszk	van	van
	2 db 5,25"-os floppy diszk	van	ajánlott
	2 db 5,25"-os floppy diszk 1 db CM 5400 típusú diszk	speciális felhasználáshoz	—
	1 db merevlemezes tár és 2 db 8"-os floppy diszk	—	ajánlott speciális felhasználáshoz
	2 db merevlemezes tár 2 db 8"-os floppy diszk	—	speciális felhasználáshoz
	1 db merevlemezes tár és 2 db 5,25" slimline floppy diszk	—	ajánlott speciális felhasználáshoz

K = Központi egység

Alapozás IV.

Addig nem mondhatjuk valamiről, hogy ismerjük, amíg nem tudjuk, hogyan működik, azaz mikor hogyan viselkedik. Operátorainkról (rabszolgáinkról, robotjainkról) ezt még nem minden vonatkozásban mondhatjuk el. Lássunk neki most e fontos feladat megoldásának!

A működés fogalma

Hogyan működik tehát egy operátor? Kellemetlen, de erre a kérdésre nem lehet általános érvényű választ adni. Egy (jó) operátor úgy működik, ahogyan működése definiálva van. Az operátorhoz tehát elválaszthatatlanul hozzátartozik az operátor működésleírása, működésének pontos definiálása. Ezért sohase elégedhetünk meg az operátor puszta nevével, bármilyen közismert névről legyen is szó. (Ez kb. annyit ér, mint egy rajz jelmagyarázat nélkül.)

Amikor működésről, viselkedésről beszélünk, mindig jelenségekről van szó. Jó, ha tudatosítjuk, hogy ez általános érvényű igazság, más ugyanis nem történhet. A jelenség tehát kulcsfogalom. Az ismeri egy rendszer működését, aki a szóban forgó rendszer viselkedésével kapcsolatban minden jelenséget ismer, e jelenségek közötti kapcsolatokkal (amik szintén jelenségek) együtt. Jelenségek tudományos megadásának, leírásának eszköze a kijelentés. (Például: „x növekszik", „y nem változik", „Péter, ha helyesen gondolkodik, meg tudja valósítani a legjobb elképzelést".) Minden jelenség időben zajlik. Így tehát minden jelenség folyamat. Folyamat, azaz valamilyen jellemzők időbeli viselkedése, időbeli alakulása. Ha tehát jelenséget írunk le (azaz jelenség modelljét készítjük el), nem tudunk mást csinálni, mint valamilyen folyamatleírást, folyamatdefiniálást.

Egy rendszer működésének, viselkedésének ismerete matematikailag csak relációk tudását jelentheti, semmi mást. Viszont minden reláció kijelentés, és minden kijelentés reláció. Jelenségek közötti kapcsolatok is kijelentések, és így relációk is. E tudás tehát kijelentések és kijelentésekre vonatkozó kijelentések tudását jelenti. Más szóval relációk és relációk közötti relációk ismeretét. (És minden reláció felfogható kijelentésekre vonatkozó kijelentésnek.)

Összefoglalva: a tudományos gyakorlat számára minden anyagi jelenség megadása, leírása folyamatmegadást, folyamatleírást, azaz valamilyen időfüggvény megadását, illetve megszerkesztését jelenti. Amikor tehát egy rendszer működését akarjuk feltárni, akkor folyamatokat kell megfigyelnünk és folyamatok közötti kapcsolatokat, ugyanis minden jelenség folyamat. Munkánk sikere attól függ, hogy jól tudjuk-e megválasztani, illetve definiálni a folyamatokat.

Kapcsolatok folyamatok között

Egyetlen jellemző (hacsak nem egy nagyon körültekintően definiált például vektorjellemző) időbeli alakulásának megfigyelése általában csekély értékű szokott lenni. Sok jellemzőt kell megfigyelnünk a köztük levő kapcsolatokkal együtt. Ez a gyakorlatban események regisztrálása útján történik. (Az esemény ugyanazt jelenti, mint a jelenség, talán azzal az árnyalati különbséggel, hogy az esemény általában jobban körülhatárolt, egyes vonatkozásokban előre ismert jelenség. Az eseménynek az a felfogása azonban hibás, mely szerint az esemény egy pillanatnyi állapot, azaz egy nulla időtartamú jelenség volna. Igazolja ezt a sportesemény példája is.)

Folyamatok közötti kapcsolatok megfigyelése azonban általában nehéz. Főleg két ok következtében. Először, mert a jellemzők (változók) száma nagy. Másodszor pedig, mert a fehér hollónál is ritkább (ha egyáltalán van) olyan jelenség, amely hat egy másikra, de a másik nem hat erre. Más szóval két rendszer között egyirányú, visszahatásmentes kapcsolat van.

Korunk matematikai eszközei azonban lényegében csak egyirányú hatások kezelésére alkalmasak, kölcsönhatások matematikai modellezésére jelenleg alig vagyunk képesek. (A jelenlegi geometriai jellegű skalár-változós „explicit" matematika nem alkalmas a más jellemzőkkel jobban megragadható „többtestproblémák" egyes változókra vonatkozó explicit megoldásának kényelmes előállítására.)

A számológépben zajló folyamatok megfigyelésében és megértésében azonban az említett nehézségek nem akadályoznak. Az első azért nem, mert a számológép úgy van szerkesztve, hogy egyszerű, könnyen áttekinthető, kevésváltozós elemekből épül fel. A kölcsönhatás pedig mint nem kívánt, káros jelenség oly mértékig vissza van szorítva, hogy elhanyagolhatónak tekinthető. Így tehát a számológépben zajló valódi jelenségek esetében joggal használhatjuk a vezérlés fogalmát. Azaz mondhatjuk, hogy egy jelenség kivált egy másik jelenséget, egy rendszer működése vezérli egy másik rendszer működését.

Visszatérve rabszolgáinkhoz, illetve robotjainkhoz, példájukon jól szemléltethető a visszahatásmentesség. Tegyük fel, hogy egy – sokkarú – rabszolga úgy indítja a másik munkáját, hogy egyik kezével felemeli a másik „indító karját", és az csak akkor indul, ha ez megtörtént. Igen ám, de ha az lusta, akkor nem engedi karját magasba emelni, sőt, igyekszik a másik ilyen irányú tevékenységét meghiúsítani. A birkózás eredménye – az erő- és energiaviszonyoktól függően – később indulás vagy nem indulás is lehet.

E tipikus visszahatásos esettel szemben áll az ideális robotok viselkedése. Ezek az indulást eredményező állapot (jelenség) kialakulásával szemben semmiféle ellenállást nem tanúsítanak. Az indulást kiváltó esemény kizárólag az indító viselkedésétől függ, az indítottétól nem. Úgy képzelhető, hogy az indítandó úgy figyeli az indítót, úgy van kapcsolatban az indítóval, hogy az indító viselkedését ezzel a megfigyeléssel, ezzel a kapcsolattal semmilyen mértékben és semmilyen szempontból nem befolyásolja (egy adott időszakra vonatkozóan).

Ezek után rátérhetünk az operátoraink működésével kapcsolatos részletesebb vizsgálatokra.

Működés és nem működés

Mindjárt az elején kijelentjük, hogy mindig minden rendszert (berendezést) állandóan működésben levőnek tekintünk. Nincs tehát olyan rendszer (berendezés), amely nem működik, legfeljebb csak olyan van, amely nem rendeltetésének, nem céljainknak megfelelően működik.

Ebben a kérdésben sem a hétköznapi, sem a tudományos szóhasználat nem következetes. (Gyakran halljuk álló gépre azt mondani, hogy nem működik. De ha például egy üzemképes, de álló felvonóra mutatva azt kérdezzük, hogy „Működik?", az lesz a válasz, hogy „Igen."

Felfogásunk szerint egy álló tárgyra nem azt kell mondanunk, hogy nincs sebessége, hanem azt, hogy a sebessége zérus.

Ember által, értelmes, valamilyen célra készített berendezés (ilyenek például operátoraink) esetében a rendszer viselkedését mindig megtervezik, a rendszer működésének mindig terv szerintinek (program szerintinek) kell lennie. (Természetesen más rendszerek esetében is értelmezhetők tervek, programok.) Így tehát minden működéshez hozzátartozik az is, hogy a működés milyen terv, milyen program szerinti. Ezért tulajdonképpen soha sincs puszta működés, mindig csak valamilyen program szerinti működés van. Ezért ha például kikapcsolunk egy gépet, akkor az nem megszűnik működni, hanem a kapcsoló (programkapcsoló) „KI" állásának megfelelően működik, viselkedik. Mégpedig azért, mert valamilyen szempontból akkor épp ez a számunkra a körülményeknek megfelelő, legelőnyösebb viselkedése, ezt a viselkedését használjuk, ezt hasznosítjuk, ez számunkra a leghasznosabb.

A működéstörténet

A működéstörténet egy ténylegesen végrehajtott vagy elképzelt megfigyelés eredménye, regisztrátuma. A gyakorlatban mindig véges időtartományra vonatkozik. Elméleti esetekben lehet végtelen időszakra vonatkozó is.

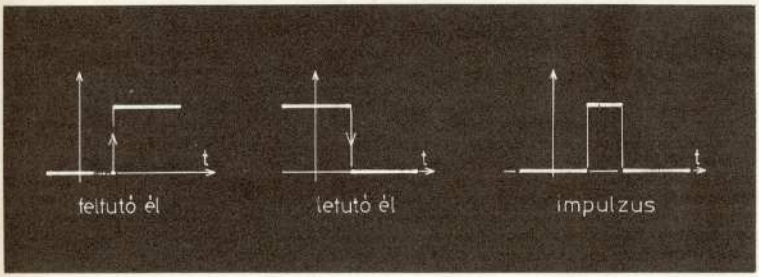
Lényegében egyes rendszerjellemzők időbeli alakulását, pillanatnyi értékeit gyűjtjük, rendezzük a működéstörténetbe. Hasznosíthatósága széles körű. Egzakt működéstörvény-leírásokban például nélkülözhetetlen. A működéstörvény lényegében arról informál, hogy mikor milyen működéstörténetek fognak előfordulni. A működéstörténet a gyakorlatban időtől függő függvények összessége, esetenként ezek ábrájával együtt.

A működéstörvény

Mindenekelőtt azt kell pontosan leírnunk, hogy mivel „táplálhatjuk" rabszolgáinkat, miket juttathatunk robotjaink kezébe. „Indító" kezük

ilyen szempontból ugyanolyan fontos, mint a többi kezük. Mint már tudjuk, az indítás tulajdonképpen csak működésmód-vezérlés, semmi több, hatására más működésmódba kerülnek, ez pedig csak azt jelenti, hogy működésük eredménye módosul. „Indító" kezükkel – ami tulajdonképpen ugyanolyan, működési eredményt kialakító információt vesz fel, mint a többi – figyelik az „indító jelet". E jel egy esemény lezajlása. Ez az esemény elvileg bármilyen lehet. Nincs akadálya például annak sem, hogy időről időre más legyen az „indulást" kiváltó folyamat. A gyakorlat azonban törekszik az egyszerű, könnyen és mindig ugyanúgy reprodukálható jelenségek alkalmazására. Ez az oka annak, hogy a tipikus „indító" jelenségek száma viszonylag kevés. Néhány a leggyakoribbak közül: 0-ból 1-be váltás („felfutó él"), 1-ből 0-ba váltás („lefutó él"), impulzus (lásd az *ábrát*).

Még egyszer hangsúlyozzuk, hogy az indító jel nem a működést indítja el, hanem csak működésmód-változást okoz. Hatására nem a működés indul meg, hanem egyes, számunkra akkor előnyös folyamatok. Vagy még pontosabban: az indító jel hatására esetleg másképp, e jel által megfelelőképp befolyásoltan folytatódnak a folyamatok.



Az indító jelet figyelő, érzékelő kéz szerepe matematikailag ugyanolyan, mint a többi kapó kézé. Kitüntetetté csak mi tesszük, általunk ismert funkcionális megkülönböztetések céljából.

Problematikus lehet az indító (a működésmód-vezérlő) kézbe kerülő indító jelek időbeli egymásutánja, ismétlődése. E kezeket érzékelőknek fogva fel, azt kérdezhetjük, hogy mi történik ismételt indítási igény érzékelések esetében, olyankor, amikor a rendszer még éppen dolgozik. Általános szabály nincs. Ez attól függ, hogy hogyan, milyen célra tervezték meg a robotot. Nyilvánvalóan van eset, amikor az a célszerű, ha ezek az igények (legalábbis bizonyos mennyiségű belőlük) nem vesznek el, és egymás után érvényesülnek valamilyen szabályok szerint (mint például a taxirendelést telefonon felvevő hivatalnok munkája során).

Más rendszerek viszont olyanok, hogy csak bizonyos („szabad") állapotban képesek érzékelni indító jelet, illetve a „foglalt" állapotban érzékeltek nem hatnak „indítólag", olyanok, mintha nem is lettek volna. (Úgy, mint például a hazai városi telefonközpontoknál a foglalt állomás hívása, azaz a beszélgetés „indítása".)

A „kész" jelről ugyanazt mondhatjuk el, mint az „indító" jelről. A „kész" jel nem a rendszer működésének megszüntéről tudósít, hanem egy működésjellemző-változásról, arról, hogy egyik működésmódból egy másikba átmenet jött létre.

A „kész" jelet sok mindenre használhatjuk. Összetett rendszerekben közvetlenül vagy átalakítva használhatók „indításra", azaz működésmód-vezérlésre. Ez után az előkészítés után rátérhetünk a működéstörvény konkrét megadásának kérdésére.

A működést a kapó kezekbe kerülő folyamatok és azok feldolgozásának módja határozza meg. Így tehát minden kapó kéz egyformán szerepet kap az adó kezekből kikerülő folyamatok alakításában. Amikor működéstörvényről beszélünk, ez mindig a rendszerrel kapcsolatos folyamatok közötti összefüggéseket jelent, mégpedig olyanokat, amelyek az adó kezekben megjelenő folyamatok egyértelmű meghatározását lehetővé teszik. Ez a működéstörvény fogalma egyes operátorok esetében. Ha azonban egy „operátorcsapatról" van szó, a helyzet kissé bonyolultabb.

Bármely operátorcsapat egyetlen operátorként is felfogható. Ez teljesen nyilvánvaló. Ily módon tehát van egy biztos eszközünk a csapat működéstörvényének leírására. A gyakorlat azonban használ egy más módszert is. Eszerint külön kell leírni az egyes komponens operátorok működéstörvényeit, aztán pedig ezeknek az operátoroknak az egymással való kapcsolatát. Nyilvánvalóan a két eredménynek nem szabad eltérnie egymástól.

Van azonban a kérdésnek egy harmadik – szintén gyakorlati – megoldása is. Ez egyes folyamatok önkényes, de valamilyen szempontból hasznos kiemelésével, középpontba állításával dolgozik. E folyamatokkal ilyenkor mesterségesen valamiféle főszerepet játszatunk. Tudjuk, hogy matematikailag ilyen főszerep nincsen. Ha viszont főszerep definiálásával könnyebbé tesszük a megértést, érdemes ezen az úton járni. E főszereppel felruházott folyamatok általában azok, amelyeknek az egész csapat munkájának vezénylését tulajdonítani lehet.

Nagyon fontos megjegyeznünk, hogy nem minden operátorcsapat olyan, hogy definiálható működésében egy könnyű áttekintést biztosító, rendező funkciójú főfolyamat. Akkor azonban, amikor ilyen létezik, nagymértékben megkönnyítheti a működéstörvény leírását. Különösen olyan esetekben hasznos ez a módszer, amikor vannak olyan változók, amelyek csak kevés értéket vehetnek fel. Ha például minden operátornak van legalább egy olyan kapó keze, amelybe csak 0 vagy 1 érték kerülhet, ezt esetleg érdemes felhasználni például „működik", „nem működik" üzemmód (azaz helyesen „1-es üzemmódban működik" és „nem 1-es üzemmódban működik" üzemmód) definiálására.

Vannak operátorcsapatok, amelyekben minden operátor teljesen egyenrangú, feladatukat csak erőltetve tudjuk vezérlési szempontból megkülönböztetni. Más esetekben világosan elkülöníthető egy vagy több olyan operátor, amelynek a többinél nagyobb számú más operátorral van közvetlen kapcsolata. Ezek működése nyilvánvalóan az átlagosnál több más operátor működését befolyásolja közvetlen módon. Az ilyen operátorok folyamatai általában alkalmasabbak szoktak lenni arra, hogy segítségükkel a többi folyamatot, a többi jellemző alakulását meg tudjuk magyarázni.

Operátorok működéstörvényével kapcsolatban még két fontos dolgot kell említenünk.

Az első – ami az eddigiek alapján már nem meglepő – az, hogy vannak olyan operátorok, amelyeknek a matematikai szempontból amúgy is egyenrangú karjai között semmiféle megkülönböztetést nem teszünk. Ezeknek az operátoroknak tehát nincs indító karjuk. („Kész" jelet – jelenséget – adó karjuk, ettől függetlenül, természetesen lehet, ha ilyenre szükség van.) Meg kell még jegyeznünk, hogy elvileg egy operátor bármely kapó keze tekinthető működésmód-vezérlő funkciójának. Sőt, olykor az is célszerű, hogy több kapó kart együtt kezeljünk működésmód-vezérlőként.

A másik fontos megjegyzés az, hogy a működésmód-vezérlésnek nemcsak egy, hanem számtalan más módja is lehet. Egy, a gyakorlatban széles körben alkalmazott megoldás nem használ „indító" jelet és „kész" jelet, hanem csak „működésben tartás" jelet. Ezzel kapcsolatban természetesen újból hangsúlyozni kell, hogy ez valójában nem működésben tartásról gondoskodik, hanem csak valamilyen típusú működésben tartásról. Ennek megszüntével a működés nem szűnik meg, hanem más módon folytatódik. Az ilyen módon vezérelt operátornak természetesen nincs indító jelet érzékelő keze és nincs „kész" jelet adó keze. Az indító jelet érzékelő kézre nyilvánvalóan nincs szükség.

A „kész" jelet adó kéz esete bonyolultabb. Ha az adott típusú működésben tartás folyamán az operátor éppen hogy elkészül a feladatával, akkor a működésben tartó jel megszünte felhasználható „kész" jelként is. Ha azonban a feladat elkészülése nem esik egybe a „működésben tartó" jel megszűnésével, akkor nincs jogunk a „működtetés" végét feladatelkészülésként jelenteni. Ilyenkor tehát értelme lehet egy külön „kész" jelet adó kar használatának is.

Operátorcsapat működésének összehangolása

A számítógép is és egy program is, lényegét tekintve, operátorcsapat. Működéstörvényeik megismeréséhez, leírásához az operátorok működésének összehangolása, koordinálása nélkülözhetetlenül fontos. Ilyen jellegű kérdésekkel következő számunkban foglalkozunk.

POGÁNY CSABA

PROGRAMOZÁSI FOGÁSOK

Sorozatunkban ötleteket, módszereket adunk közre azok számára, akik már megismerkedtek mikrogépükkel és/vagy azt munkájukban is használják. Reméljük, hogy e sorozat írásába ők is bekapcsolódnak; beküldött programozási fogásaikat – amennyiben valóban ötletesek – szívesen közöljük.

BASIC interpreterek – a program változói

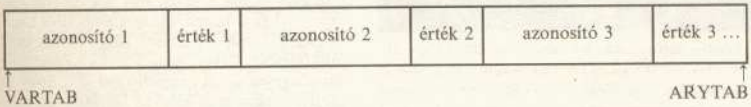
Az előző számban megkezdtük annak ismertetését, hogyan használják a BASIC munkaterületet az interpreterek. A továbbiakban a munkaterületnek arról a részéről lesz szó, amelyben a BASIC programok futásuk közben a változók értékét tárolják. Előbb azonban két további megjegyzést teszünk a munkaterület „felosztását" definiáló mutatókról.

Ezek a mutatók (TXTTAB, VARTAB stb.) általában csak a PEEK, POKE utasításpár segítségével állíthatók elő, illetve változtathatók meg. Kivételt jelent a HT iskolaszámítógép, amely lehetővé teszi a NEMSIZ és a STRTAB mutató állítását. A gép bekapcsolása után a READY? felirat jelenik meg. Válaszként a BASIC munkaterület végét kell decimális alakban leírni, majd a [NEWLINE] billentyűt megnyomni. Ha nem írunk be semmit, NEMSIZ automatikusan NEMTOP lesz.

A HT BASIC interpreterének külön utasítása van a sztringterület aljának beállítására. Erre a CLEAR <aritmetikai kifejezés> alakú utasítás szolgál. Az utasítás töröl minden változót és a STRTAB értékét úgy állítja be, hogy a sztringterület nagysága az <aritmetikai kifejezés> értékével egyezzen meg. Ebben az esetben természetesen szükség van egy további mutatóra, amely a sztringterület első szabad bájtjára mutat. Ez a 16598–16599-es címeken található.

Változók használata

Az egyszerűség kedvéért a tömbök tárolásával most nem foglalkozunk. A különféle interpreterek a változókat a következő séma szerint tárolják:



Egy változó az őt azonosító és az értékét megadó részből áll. Az azonosító megadja a változó *nevét és típusát*, míg az *érték* a változó utolsó értékét tárolja az interpreterre jellemző formában. Amikor egy LET Y = X+1 értékadás kerül végrehajtásra, az interpreter végigolvassa a változóterületet és megkeresi az X nevű valós változó azonosítóját. Ha megtalálja, akkor az azonosító után található érték felhasználásával kiszámítja az (X+1)-et, és ezt az értéket az Y azonosítója után beírja.

Ha az interpreter az X azonosítóját nem találta meg, kétféleképpen folytathatja a munkáját. X azonosítóját valamilyen kezdőértékkel (például 0) beírja a változóterületre és azzal számol tovább (C–64, HT), vagy egyszerűen hibát jelez (ZX81).

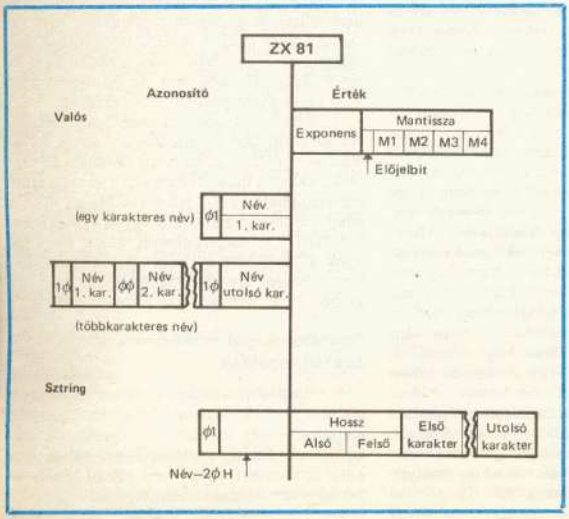
A változók értékének tárolását a legtöbb interpreter azonos logika szerint végzi, az azonosító rész azonban általában más és más elvek alapján épül fel. A legegyszerűbb – és ezért talán a leghatékonyabb – tárolási eljárást a HT interpretere valósítja meg. A ZX81 tárolását bonyolítja, hogy *akármilyen* hosszú neveket megenged valós változók jelölésére.

A táblázataink összefoglalják, hogy a ZX81, a C–64 és a HT interpreterei hogyan is tárolják az egyszerű változókat (*1., 2., 3. táblázat*).

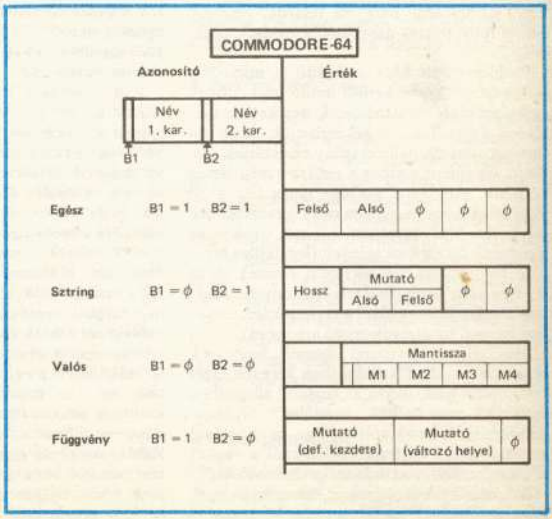
```
10 X=0.0
20 XX=VARPTR(X)
30 INPUT X
40 PRINT "X=";X;" ";
50 FOR I=0 TO 3
60 PRINT PEEK(XX+I)
70 NEXT I
80 PRINT " "
90 GOTO 30
```

1. program

1. táblázat



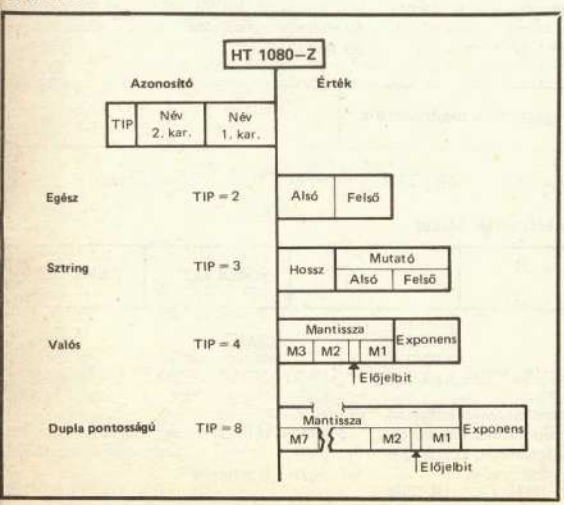
2. táblázat



```
100 REM KIPROBALASA: RUN 400
110 :
120 REM ************************
130 REM *                      *
140 REM *                      *
150 REM *   VARPTR C-64 -RE    *
160 REM *                      *
170 REM *                      *
180 REM *  HIVASA:             *
190 REM *                      *
200 REM *  X$=<NEV>:GOSUB 100  *
210 REM *  (X%-BAN A MUTATO)   *
220 REM *                      *
230 REM ************************
240 :
250 REM A VALTOZOTERULET ELEJE,VEGE
260 P1%=256*PEEK(46)+PEEK(45)
270 P2%=256*PEEK(48)+PEEK(49)
280 :
290 REM CIKLUS ELEJE
300 X%=P1%
310 :
320 REM NEV ES TIPUS ELOALLITASA
330 N1=PEEK(X%):N2=PEEK(X%+1):B1=0:B2=0
340 IF N1>=128 THEN N1=N1-128:B1=1
350 IF N2>=128 THEN N2=N2-128:B2=1
360 N$=CHR$(N1)
370 IF N2<>0 THEN N$=N$+CHR$(N2)
380 TY= 2*B1+B2
390 IF TY=1 THEN N$=N$+"$"
400 IF TY=3 THEN N$=N$+"%"
410 IF TY=2 THEN N$="FN"+N$
420 :
430 REM EGYEZIK-E A NEV X$-RAL?
440 IF X$=N$ THEN X%=X%+2:RETURN
450 X%=X%+7:IF X%=P2% THEN X%=0:RETURN
460 GOTO 330
470 :
480 REM FOPROGRAM A RUTIN KIPROBALASARA
490 X=12:X$="X":GOSUB 120:PRINTX%
500 X$="SD":GOSUB 120:PRINTX%
510 SE$="URY LASZLO"
520 X$="SE$":GOSUB 120:PRINTX%
```

2. program

3. táblázat




```
6000 REM ************************
6010 REM *                      *
6020 REM *  EGYSZERU VALTOZO DUMP *
6030 REM *                      *
6035 REM *                      *
6040 REM *   HASZNALT VALTOZOK:   *
6045 REM *                      *
6050 REM *    XI$(3),XL(3),X,X$   *
6060 REM *    X(3),XB,XC,XP,XQ    *
6070 REM *    X1,X2,XI,XJ,XT      *
6090 REM *                      *
6100 REM ************************
6110 PRINT"J";
6120 DATA " ",7,$,7,FN,7,%,7
6130 REM XI$(3) AZ EGYES TIPUSOK JELET
6140 REM XL(3) AZOK HOSSZAT TARTALMAZZA
6150 FOR XI=0 TO 3
6160 READ XI$(XI),XL(XI)
6170 NEXT XI
6180 :
6190 REM X(0) A VALTOZOTERULET JELENLEGI VEGE
6200 REM ES EGYBEN AZ X VALTOZO KEZDETE
6210 X(0)=256*PEEK(48)+PEEK(47):X=0.0
6220 REM X(1) AZ X$ HELYE
6230 X$="URY":X(1)=X(0)+XL(0)
6240 REM X(3) AZ X% HELYE
6250 X%=0:X(3)=X(1)+XL(1)
6260 :
6270 REM A VALTOZOTERULET XP ES XQ KOZE ESIK
6280 XP=256*PEEK(46)+PEEK(45)
6290 XQ=256*PEEK(48)+PEEK(47):PRINT"M";
6300 :
6310 FOR XI=XP TO XQ-1
6320 :
6330 REM X1= A NEV ELSO KARAKTERE
6340 REM X2= A NEV MASODIK KARAKTERE
6350 REM A 7. BITEK A TIPUST KODOLJAK
6360 REM XT= A TIPUS KODJA
6370 X1=PEEK(XI):X2=PEEK(XI+1):XB=0:XC=0
6380 IF X1>=128THEN X1=X1-128:XB=1
6390 IF X2>=128THEN X2=X2-128:XC=1
6400 XT=2*XB+XC
6410 :
6420 REM XNV=A VALTOZO NEVE
6430 XNV$=CHR$(X1)+CHR$(X2)+XI$(XT)
6440 IF X2=0 THEN XNV$=CHR$(X1)+XI$(XT)
6450 XNV$=LEFT$(XNV$+"      ",3)+"="
6460 IF XT=2 THEN GOTO 6540
6470 :
6480 REM A VALTOZO ERTEKET A MEGFELELO TIPUSU
6490 REM X-NEVU VALTOZOBA MASOLJUK
6500 REM MAJD AZT A VALTOZOT KIIRJUK
6510 FOR XJ=2 TO XL(XT)-1
6520 POKE X(XT)+XJ,PEEK(XI+XJ)
6530 NEXT XJ
6540 XI=XI+6
6550 ON XT+1 GOSUB 6570,6580,6590,6600
6560 NEXT XI: RETURN
6570 PRINT XNV$;X:RETURN
6580 PRINT XNV$;CHR$(34);X$;CHR$(34):RETURN
6590 PRINT XNV$;" FUNCTION":RETURN
6600 PRINT XNV$;X%:RETURN

READY.
```

3. program

A táblázatokból jól látszik, hogy a valós számokat mennyire hasonlóan tárolják: külön az exponens és külön a mantissza részt. Különbség csak a mantissza hosszában (azaz az értékes jegyek számában) és a bájtok tárolási sorrendjében van. Az interpreterek a mantisszát, az exponenst és az egészeket – megfelelő hosszú – előjeles, kettes számrendszerbeli számnak tekintik, és kettes komplemens kódban tárolják.

Lényeges eltérés csak a ZX81, illetve a HT és C–64 interpreterek sztringtárolása között van. A ZX81 a sztring azonosítója után a sztring hosszát, majd a sztring karaktereit tárolja értékként.

A másik két számítógép csak a sztring hosszát és egy kétbájtos mutatót tárol. A mutató a sztringterületen tárolt karaktersorozat első bájtjára mutat. Azokban az esetekben, amikor a sztring a program szövegében is megtalálható, a mutató a programba mutat vissza. Ez a helyzet például az X$="ABCD" értékadás után.

A változók elhelyezésére vonatkozó információink hasznosításához már csak azt kell tudnunk, *hol* vannak az *egyes* változók. A HT esetében könnyű a dolgunk, a VARPTR függvény minden egyes változóra megadja a változóhoz tartozó érték első bájtjának a címét. Ha ilyen függvény nincs, külön rutint kell írnunk, amely megkeresi az adott változóhoz tartozó azonosítót a memóriában. Ha azonban nincs a BASIC munkaterületen tárolt program, a TXTTAB ismeretében pontosan tudhatjuk, hol van a változónk.

Ha a következő parancsokat egymás után végrehajtjuk, akkor válaszul 1-et kapunk. A második paranccsal ugyanis az A változó azonosító részét átírtuk B-re, így a gép úgy érzi, hogy B-t tároltuk:

```
HT      LET A=1
        POKE VARPTR (A) - 1,66
        PRINT B
C-64    LET A=1
        POKE 2051,66
        PRINT B
```

Ez azonban csak játék. A következő példa – HT gépre – két sztringváltozó értékét cseréli fel anélkül, hogy egyetlen sztringműveletet végrehajtottunk volna:

```
10  A% = VARPTR (A$):B% = VARPTR (B$)
20  POKE A% - 1, ASC („B"):POKE B% - ASC („A")
```

A fenti program lényegesen gyorsabban hajtódik végre, mint az

X$ = A$:A$ = B$:B$ = X$

utasítássorozat! Azonkívül a sztringterületen nem jelenik meg a teljesen felesleges X$ változó. Ha például LEN (A$)=120, akkor ezzel az eljárással 360 bájt helyet spórolunk meg!

A fenti trükk főleg mátrixműveletek esetén hasznos, amikor 100 vagy 1000 elem mozgatása helyett mindössze 3 bájtot cserélünk meg.

A VARPTR segítségével például megvizsgálhatjuk, hogyan tárolja az interpreter a BASIC változókat. Hajtsuk végre az *1. programot*. Az eredmény (a feltüntetett X értékek esetén):

```
990 REM LOCAL RUTIN
1000 POKE 12*4096,PEEK(45):POKE 12*4096+1,PEEK(46)
1010 POKE 12*4096+2,PEEK(47):POKE 12*4096+3,PEEK(48)
1020 POKE 12*4096+4,PEEK(49):POKE 12*4096+5,PEEK(50)
1030 POKE 12*4096+6,PEEK(51):POKE 12*4096+7,PEEK(52)
1040 POKE 45,PEEK(49):POKE 46,PEEK(50)
1050 POKE 47,PEEK(49):POKE 48,PEEK(50)
1060 RETURN
1989 :
1990 REM GLOBAL RUTIN
2000 POKE 45,PEEK(12*4096):POKE 46,PEEK(12*4096+1)
2010 POKE 47,PEEK(12*4096+2):POKE 48,PEEK(12*4096+3)
2020 POKE 49,PEEK(12*4096+4):POKE 50,PEEK(12*4096+5)
2030 POKE 51,PEEK(12*4096+6):POKE 52,PEEK(12*4096+7)
2040 RETURN
```

4. program

X = 0	0	0	32	0
X = −1	0	0	128	129
X = 0,5	0	0	0	128
X = 0,25	0	0	0	127

Hasonlóképpen járhatunk el a többi típus esetén is.

VARPTR a C–64-en

A fentiek ismeretében könnyen elkészíthetjük a C–64-en hiányzó VARPTR függvényt. A következő alprogram X$-ban várja az (egyszerű) változó nevét a típusmegjelöléssel együtt. A visszatéréskor X% tartalmazza a változó értékét megadó első bájt címét.

Ha X%=0, az azt jelenti, hogy a változót még nem használták (*2. program*).

A program változóinak kilistázása (változó 'dump')

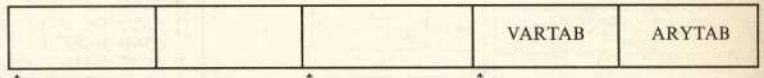
Következő példánk nemcsak az eddigi ismeretek gyakorlására alkalmas, hanem igen hasznos programfejlesztési segédeszköz is. Olyan alprogramot ismertetünk, amely a program addig használt változóinak értékét kiírja a képernyőre (*3. program*). Az alprogram használatához egyáltalán nem kell tudnunk, milyen változókat használt a program többi része.

A fenti alprogramot a vizsgálni kívánt program után kell beírni (vagy ha rendelkezünk a megfelelő BASIC-bővítéssel, az APPEND utasítás segítségével betölteni). Amikor csak szükségünk van a változók értékére, a GOSUB 6000 utasítást kell kiadnunk, és a képernyőn máris megjelenik a lista. Arra kell csak ügyelni, hogy programunk ne tartalmazza azokat a változókat, amelyeket az alprogram használ.

Lokális változók használata

A BASIC használata közben sok bosszúságot okoz, hogy az alprogramok írásánál figyelni kell arra, hogy a program többi része milyen változókat használ. Ez különösen akkor okoz problémát, ha mások által megírt programrészeket szeretnénk felhasználni. A következőkben két rövid alprogramot mutatunk be, amelyek lehetővé teszik, hogy az alprogramjainkban tetszőleges lokális változókat használjunk. Az első alprogram előtt a BASIC munkaterület így néz ki:

Az első szubrutin után a helyzet így alakul:



A második szubrutin azután visszaállítja az eredeti állapotot. Ha lokális változókat szeretnénk használni egy alprogramban (kezdőcím JJJJ), akkor azt így kell meghívnunk:

```
GOSUB 1000 : GOSUB JJJJ : GOSUB 2000
```

(*4. program*).

Végül a HT gépen egy lokális változókat használó és 'dump'-ot készítő alprogramot mutatunk be (*5. program*).

DR. ÚRY LÁSZLÓ

```
1 REM A PROGRAM ILYEN ALAKBAN FUTTATVA
2 REM A 10-20. SOROKBAN LEVO ERTEKEKET IRJA KI.
3 REM A PROGRAM LEFUTASA UTAN AZ
4 REM ALPROGRAM VALTOZOINAK ERTEKE
5 REM ELVESZIK.
10 A=12.34:A%=23:A#=2323323232.545
20 B=12E3: C2#=3.14159265
30 :
40 :
100 REM ************************
110 REM *                      *
120 REM *                      *
130 REM * EGYSZERU VALTOZO DUMP *
140 REM *                      *
150 REM *      HT-1080Z -N     *
160 REM *                      *
170 REM ************************
180 REM LOCAL RUTIN
190 REM A MUTATOK ERTEKET EGY HASZNALATLAN
200 REM RESZRE MENTJUK EL.
210 POKE 16446,PEEK(16633): POKE 16447,PEEK(16634)
220 POKE 16448,PEEK(16635): POKE 16449,PEEK(16636)
230 POKE 16450,PEEK(16637): POKE 16451,PEEK(16638)
240 POKE 16633,PEEK(16637): POKE 16634,PEEK(16638):CLEAR
250 :
260 REM A DUMP-OT VEGZO RESZ:
270 REM LOGIKAILAG MEGEGYEZIK
280 REM A C-64-RE ADOTT PROGRAMMAL.
290 DATA "%","$"," ","#"
300 FOR I=0 TO 3:READ TI$(I):NEXT I
310 P1=256*PEEK(16447)+PEEK(16446)
320 P2=256*PEEK(16449)+PEEK(16448)
330 X%=0%:P(0)=VARPTR(X%)
340 X$="":P(1)=VARPTR(X$)
350 X=0.0:P(2)=VARPTR(X)
360 X#=0.:P(3)=VARPTR(X#)
370 :
380 FOR I=P1 TO P2-1
390 TI=PEEK(I)
400 IF TI=2 THEN TT=0
410 IF TI=3 THEN TT=1
420 IF TI=4 THEN TT=2
430 IF TI=8 THEN TT=3
440 :
450 NEV$=CHR$(PEEK(I+2))+CHR$(PEEK(I+1))+TI$(TT)+"="
460 FOR J=1 TO TT
470 POKE P(TT)+J-1,PEEK(I+J+2)
480 NEXT J
490 PRINT NEV$;
500 ON TT+1 GOSUB 540,550,560,570
510 I=I+TI+2
520 NEXT I: GOTO 600
530 :
540 PRINT X%:RETURN
550 PRINT X$:RETURN
560 PRINT X :RETURN
570 PRINT X#:RETURN
580 :
590 REM GLOBAL RUTIN
600 POKE 16633,PEEK(16446): POKE 16634,PEEK(16447)
610 POKE 16635,PEEK(16448): POKE 16636,PEEK(16449)
620 POKE 16637,PEEK(16450): POKE 16638,PEEK(16451)
630 STOP
```

5. program

Kabalapályázatunkra érkezett

2. *Kókai Tibor, Dunakeszi*

FELHÍVÁS

A Neumann János Számítógéptudományi Társaság Csongrád megyei Szervezete és Orvosbiológiai Szakosztálya ez év december 3–5. között rendezi meg Szegeden a 12. SZÁMÍTÁSTECHNIKAI ÉS KIBERNETIKAI MÓDSZEREK ALKALMAZÁSA AZ ORVOSTUDOMÁNYBAN ÉS A BIOLÓGIÁBAN cí mű kollokviumot, a Szegedi Orvostudományi Egyetem Számítástechnikai Központja szervezésében.

A már több mint egy évtizede rendszeresen megtartott rendezvény célja, hogy fórumot biztosítson a számítástechnika egészségügyi alkalmazásaival és a biomatematikai kutatásokkal foglalkozó hazai szakembereknek új eredményeik bemutatására és tapasztalataik kicserélésére. A kollokviumot ezúttal először hirdették meg nemzetközi részvétellel; hivatalos nyelve magyar és angol. Az angol nyelvű előadásokhoz a rendezőség szinkrontolmácsolásról gondoskodik.

Nagyfokú aktualitása miatt a kollokviumon kiemelt témaként szerepel a mikroprocesszorok, mikroszámítógépek egészségügyi alkalmazása. További témakörök: egészségügyi informatika és adatfeldolgozás, adatelemző és biometriai módszerek, biológiai jelek feldolgozása, biológiai rendszerek matematikai modellezése és számítógépes szimulációja, számítógéppel segített orvosi diagnosztika.

Jelentkezési lapok és további információk a kollokvium szervező bizottságától kérhetők a következő címen:

Dr. Győri István,
Szegedi Orvostudományi Egyetem
Számítástechnikai Központ,
6720 Szeged, Pécsi u. 4/a.
Telefon: 62/11-084

COMMODORE–64

Az év számítógépe 1983-ban

E házi fél-professzionális személyi számítógéppel a Commodore cég a VIC–20 után „igazi" személyi számítógépet dobott a piacra. Az idézőjel lényeges: a gép olyan, mint az igazi személyi számítógép, de a teljesítménye... Nos, a teljesítménye és alkalmazási lehetőségei tükröződnek az árában (áron most nem elsősorban a hazai árakat értve). Itt a hazai tapasztalatok alapján elsősorban negatív állításokat kell előrebocsátanunk. Tehát például ez a gép nem alkalmas soros kötegelt feldolgozások elvégzésére. Egész egyszerűen mondva: ha nincs pénzünk professzionális, egy- vagy többfelhasználós gépre (M08X, VT–20, TPA–1140, R22), hanem csak egy Commodore–64-re futja, akkor ne próbáljunk meg azokon a gépeken megoldható feladatokat egy az egyben Commodore–64-en megvalósítani.

A gép kinézésre kellemes benyomást kelt. Diszkrét, kéz alá simuló, esztétikusan kiképzett dobozt láthatunk, nagyon jó klaviatúrával. A tápegységet a hőstabilitás érdekében külön egységnek tervezték. Figyelmet érdemel a csatlakozók kiképzése: egy csatlakozóvezetéket sem dughatunk más csatlakozópontba. Még műszakilag képzetlen személy is képes installálni.

A BASIC interpreter

Bekapcsolás után a gép alaphelyzetben a ROM-ban levő BASIC interpretert indítja. A BASIC megvalósítása általában jó. Az utasítások végrehajtási sebessége változó: általában 1000 IPS (utasítás/szekundum) alatti értékkel számolhatunk, valós és stringműveletek esetén kb. 150–300 IPS-szel. Ez a BASIC megkülönbözteti a valós és az egész változókat, az egész aritmetika nagyon gyors a valós aritmetikához képest.

A nyelv használata viszonylag hamar elsajátítható más BASIC alapján. A megvalósítás hátrányául róható fel az interpreter belső kialakításából fakadó két jelentős sajátosság: a program módosítása esetén a változók törlődnek, illetve a szemétgyűjtés (a BASIC bizonyos adatszerkezeteinek belső kezeléséhez szükséges belső mechanizmus) megvalósítási módja. Az első jelenség a fejlesztési periódusban jelent kényelmetlenséget, mivel minden felfedezett hiba kijavítása után újra kell futtatni a programot az elejétől, a második pedig futtatás közben: a program futása néha percekre leáll, a gép ilyenkor még hardver reset segítségével sem állítható meg. Ez a jelenség a sok karaktermanipulációt végző programoknál gyakori, és végül is meg lehet szokni.

Szokatlan a be/kiviteli utasítások megvalósítása: a képernyőn kívül minden adatállomány egy uniform utasításkészlet segítségével kezelhető (open, close, input#, get#, print#). Az egyes perifériák az open utasítással címezhetők. A tényleges be/kivitel esetén a programnak „csak" be kell tartania az egyes perifériákra előírt, esetlegesen eltérő szabályokat.

A hardver architektúra

A gép jellegéből adódóan – elsősorban játékprogramok futtatására szánták – kiépítése szokatlan. A gép 6510 alapú (6502 kompatibilis). A memória 64 kbájt RAM-ból és 16 kbájt ROM-ból áll, továbbá a karakter ROM-ból, amit a gép speciálisan kezel. E választékból a programozó szabadon konfigurálhatja a memóriát, illetve egyes RAM területeket cartridge csatlakoztatásával még ezenkívül is módosíthat. A gép alaphelyzetben történő bekapcsolásakor aktív a BASIC ROM és a KERNAL ROM (8–8 kbájt), amelyek közül az előbbi a BASIC interpretert, az utóbbi az operációs rendszert tartalmazza.

A felhasználó számára rendelkezésre álló terület a $0800–$9FFF és a $C000–$CFFF (a $ jel hexadecimális értékeket vezet be). A gépben levő speciális lehetőségeket támogató IC-k: a VIC (Video Interface Controller), a SID (Sound Interface Device) és a két CIA (Complex Interface Adapter). A VIC támogatásával a szabványos 40 × 25-ös képernyőt 320 × 200-as bit mapped módúvá (1 bit = 1 képernyőpont) átállítva lehet kezelni (egy szín + egy háttérszín), vagy 160 × 200-as méretben multicolor módban. Ez az eszköz teszi lehetővé a sprite-ok vagy másképpen MOB-ok (Movable Object Block) kezelését. Egy MOB 21 × 24 képpont méretű; egyszerre max. 8 darab használható. Ezek egymástól és a képernyőn levő információktól függetlenül szabadon és igen gyorsan mozgathatók, illetve kezelhetők. Az IC-k minden lehetőségét itt nem tudjuk felsorolni, de az összefoglaló részben néhány lehetőségre utalni fogunk.

A gép hangját a SID integrált áramkör állítja elő. Ennek segítségével 3 hanggenerátort programozhatunk egymástól függetlenül vagy függő módon, 7 oktáv terjedelemben. Az egyes hanggenerátorok egymással „összeköthetők" – lehetőség van körmodulációra, külső moduláló jel bevitelére, az egyes jelalakok formálására, a hangerő szabályozására stb. Az eszköz lehetőségei szinte kimeríthetetlenek. Az átlagos programozó számára azonban az alacsony szintű programozás következtében, megfelelő segédlet nélkül nehezen kezelhető.

A két CIA-n keresztül kezelhetők a további perifériák: a kazettás magnetofon, a hajlékonylemez, a joystick, a fényceruza, a nyomtató stb.

Az alapgép nem túl gyors: 1 MHz-nél valamivel kisebb órajellel dolgozik.

A perifériák

A Commodore–64 gép moduláris koncepciójú. Ebből következően az alapgép a lehető legkevésbé köti meg a perifériák kezelését. A legfontosabb perifériák a következők:

Megjelenítő. Tv vagy video monitor. Hazai viszonylatban elsősorban tv-t szokás használni. Megjegyzendő, hogy nem kapható megfelelő minőségű, méretű és kialakítású készülék ezekhez a gépekhez. A tv helyes megválasztása igen fontos, kényelmi és munkavédelmi szempontból egyaránt. A legmegfelelőbb valamelyik VIDEOTON gyártmányú, kisképernyős készülék lenne, vagy egy kétnormás, kisképernyős, jó minőségű színes tv. Ez az a pont, ahol nem szabad a beruházást sajnálni. A gép az UHF 36. csatornán adja az RF jelet. A jel fel- és letranszformálása, illetve a környezetből származó zajok következtében a képminőség igen könnyen leromlik. Zavaró lehet a 2. adás beindulásakor keletkező intermodulációs torzítás. Ezt csökkenthetjük az alapgép trimmelésével. Nem megfelelő megjelenítő használata esetén nagyon hamar kifárad a szem.

Az ideális megjelenítő a gyári, színes és hangos video monitor lenne. A tapasztalat szerint ez felel meg legjobban a munkavédelmi követelményeknek. Kisebb igényű (színes grafika nélküli és hangjelzést nem igénylő) feldolgozásokhoz beszerzett gépekhez egyszínű monitor ajánlható. Sokkal jobb képminőséget ad, mint a tv-készülék.

Magnetofon (VC–1530). A géphez csak a Commodore gyári készüléke csatlakoztatható. A kezelési utasításban leírtak betartása mellett igen stabil periféria, de nagyon lassú.

Hajlékony mágneslemezes egység (VC–1541). Saját tápegységgel rendelkező meghajtó 5 ¼"-es lemezek kezelésére, kapacitása 170 kbájt. Ebben az egységben domborodik ki a Commodore egyik koncepciója, az intelligens periféria. Ezen eszköz körül viták dúlnak a szakemberek között. A hajlékony mágneslemezes egység soros vonalon csatlakoztatható a géphez, és ez a vonal elég lassú.

Megjegyezzük, hogy a mágneslemezegységek felfűzéses módszerrel (daisy chaining) csatlakoznak ugyanarra a soros vonalra a nyomtatóval együtt. Ez a csatlakoztatási mód viszonylag alacsony, kb. 1 kbájt/szekundum átviteli sebességet biztosít. A gyártó maximálisan 5 egység csatlakoztatását engedi meg, ami

külföldi tapasztalatok szerint nem mindig lehetséges.

A periféria tartalmazza a mechanikát, valamint egy 6502 alapú vezérlőegységet is. A periféria intelligens. Az adatátvitel a gép és az eszköz között egy közbenső, az eszközben elhelyezett tárolón keresztül történik, és jó programszerkezet esetén a vezérlés és az adatátvitel elválasztható egymástól.

Ez az egység végzi a lemezes állományok kezelését is. Az eszköz a program, a soros, a random és a relatív adatállományok kezelését támogatja. Itt kell megemlíteni, hogy nem maga a lemezegység vagy a vonal lassú, hanem a BASIC be/kimenet. Jó programszervezéssel és alapos hardverismeretekkel az átvitel hatásfoka javítható.

Sornyomtató. Többféle típust lehet kapni, elsősorban VC–1515, VC–1525, VC–1526, SEIKOSHA GP–100/VC, MPS 801, valamint elvétve EPSON RX–80, FX–80 típusokat. Az első csoportba tartozó nyomtatók közös sajátossága, hogy igen egyszerűek és olcsók. Hátrányuk, hogy nem képesek lapot dobni. Külön kiemelendő a VC–1515-ös nyomtató: ehhez 8″ széles papírt kell használni, ami gyakorlatilag beszerezhetetlen. Sokkal jobbak és megbízhatóbbak, de természetesen drágábbak a második csoportba tartozó nyomtatók. Ezek már stabilabbak és a működési sebességük is jobb. Ügyviteli célokra inkább ezek ajánlhatók.

Egyéb perifériák. A géphez közvetlenül vagy megfelelő illesztéssel egy sereg további periféria, kiegészítő csatlakoztatható. Ilyenek a botkormány, amit elsősorban játékprogramokban alkalmaznak, a fényceruza és általában az összes olyan periféria, amely a megfelelő illesztő hardverrel és illesztő szoftverrel felszerelhető. Tetszőleges, RS–232 vagy IEEE 488 szabványnak megfelelő illesztésű perifériát csatolhatunk, többek között 10 Mbájtos nagylemezt is. E lehetőségnek a gyári katalógusokban vagy a szállítók prospektusaiban nézhetünk utána.

A szoftver

A Commodore–64 szoftverparkja nagyon jól fejlett. A teljes felsorolásra nem vállalkozhatunk, csak a legfontosabb és legelterjedtebb szoftvereket, fejlesztő eszközöket, nyelvbővítéseket, fejlesztő rendszereket mutatjuk be.

HELP. BASIC-bővítés. Programfejlesztői támogatást ad: listázás, újrasorszámozás, törlés, nyomkövetés, változók megjelenítése stb.

SIMON'S BASIC. BASIC-bővítés. Saját koncepciójú fejlesztői támogatás: fejlesztői segédletek, grafikus utasítások, hanggenerálás, eljáráshívás stb. 114 új BASIC utasítást valósít meg.

SUPERGRAFIK–64. BASIC-bővítés grafikai lehetőségekkel. Ez a bővítés nem támogatja a sprite-kezelést.

Assemblerek. A két legelterjedtebb változat a Commodore gyári assembler és a DATA BECKER által terjesztett PROFIMAT fejlesztő rendszer.

AUSTROCOMP. BASIC fordítóprogram. Két menetben generál kódot. Használata javítja a végrehajtás hatásfokát, de az inkompatibilitások miatt kellő óvatosságot igényel.

FORTH. Programnyelv. Ismertetése az 1983. évi számban megjelent.

PASCAL–64. PASCAL fordítóprogram. Csapnivaló minőségű. Semmi köze a PASCAL nyelv definíciójához. Használata ellenjavalt.

TURTLE–II. A LOGO nyelv egy részhalmazát valósítja meg. Kicsit lassú, de jól alkalmazható.

CP/M. Z80 mikroprocesszorra használt operációs rendszer. Nem kompatibilis az eredeti CP/M-mel (ernyőméret, B/K bájt stb.).

Rendszerprogramok. A rendszerprogramok alapkészlete a mágneslemezes egységhez adott DEMO DISK-en található. A többi szabadpiaci forgalomban kapható. Ilyen programok: editorok (sprite, karakter, kép, szöveg stb.), másolóprogramok, monitorok, javítóprogramok, diagnosztizáló programok stb.

Ügyviteli alkalmazások. Számos különböző célú, speciális és általános célú program kapható.

Játékprogramok. Ezek választéka a leggazdagabb. Bár nem becsüljük le a számítástechnikai kultúra terjesztésében betöltött szerepüket, ismertetésükre a cikk eltérő célja és jellege miatt itt nem térünk ki.

A szoftverpark értékelése

A belföldön beszerezhető szoftverek választéka a nyugati piacokhoz képest igen szűk. Az egyes termékek ára és használati értéke is nagyon eltérő. Mivel a belföldön használt Commodore–64 gépek döntően nem természetes, hanem jogi személyek birtokában vannak, igen eltérőek a követelmények, így a külpiaci beszerzés nem képes az igényeket kielégíteni. Szükség van a hazai fejlesztésre, ami a tapasztalatok szerint most van kialakulóban. Számos fejlesztő cég dolgozik már, sőt párhuzamos fejlesztések is vannak. A piac szerveződése azonban igen lassan folyik.

Üzemeltetés

A gép szobahőmérsékleten üzemeltethető. Kerülni kell a hajlékonylemezek toronyként való kiépítését. Általában minden egység számára biztosítani kell a megfelelő természetes szellőzést. Hőhatása miatt kerülendő az erős természetes vagy mesterséges fény. Túlzottan hosszadalmas feldolgozások, folyamatos hajlékonylemez-használat esetén a gép hibázni kezd. Ilyenkor pihentessük. Figyeljünk a környezet pormentességére is.

Az alkalmazások helyzete

E gép elterjedése igen sok vállalatnál meghozta a számítástechnika-alkalmazási kedvet. Kisebbfajta forradalom zajlik éppen. A gépek terjedésével azonban magasabb szinten ismétlődni látszik az egyszer már lezajlott válság: túl sok a szakmailag megalapozatlan igény és túl sok az átgondolatlan gépvásárlás. Az állami csatornákon importált gépek, ugyanúgy, mint a „magánimportban" behozottak, elsősorban vállalati alkalmazásba kerülnek. Utaltunk rá, hogy az elterjedt nyomtatótípusok nem, vagy csak nehézkesen alkalmasak adatfeldolgozási célokra.

A gépek beszerzésével egyidőben terjedőben van a hardvergyártó által sugalmazott hurrá optimizmus: „Ez olyan egyszerű, hogy akár még te is meg tudod tanulni!" Meg, de nem mindegy, hogy milyen szinten és mennyi idő alatt.

El kellene kerülni azt is, hogy a gépet a méretét és teljesítményét jelentősen meghaladó, nagy alkalmazások céljaira vásárolják meg egyes felhasználók (nagy adatbázisok on-line kezelésére, VT–20-ba sem beférő kötegelt feldolgozásokhoz stb.). Az ebből fakadó későbbi kudarcélmények hátráltatják a már meglevő eszközpark hatékony kihasználását és a számítástechnika-alkalmazás, valamint a számítástechnikai kultúra terjedését.

Több, különböző szintű (vállalati, ágazati) döntés született, de bizton állítható, hogy az elkészült programok számára nincs még meg a megfelelő nyilvános szoftverpiac. Pedig szükség lenne rá. Nagyon.

Értékelés

A Commodore–64 személyi számítógép a maga nemében kiváló. A professzionális gépek között is megállja a helyét, ha megfelelően kiépített gépet, megfelelő szoftverekkel, megfelelő módon használnak. Ezzel a géppel nem tudunk egyszerre sok adatot (néhányszor száz kbájt) megbízhatóan kezelni. Előnye viszont, hogy olcsóbb, mint a professzionális gépek. Az (elméletileg és) gyakorlatilag beszerezhető hardver- és szoftverelemek a számítástechnika alkalmazásának teljesen új területét nyithatják meg. A géphez beszerezhető dokumentációk általában jók, átlagos mennyiségű hibával tarkítottak.

A hardverlehetőségeket és teljesítőképességet a jelenlegi szoftverek nem használják ki. Az igazán hatékony szoftverek kifejlesztéséhez azonban még hiányoznak a megfelelő fejlesztő eszközök.

A gép kezelése a felhasználó számára általában semmiféle nehézséget nem jelent. Az alapok órák alatt elsajátíthatók. Figyelemmel kell lenni azonban a részletekre is, mert különben a gép igen hamar használhatatlanná válhat. A csatlakozások kiépítése olyan, hogy a sűrű szét- és összeépítés a vezetékeket és a csatlakozókat tönkreteheti. A folyamatos üzemű lemezírás és -olvasás esetén a szellőzés nem elegendő.

A gépre történő fejlesztéskor, adaptáláskor különös tekintettel kell lenni a korszerű módszertanra és a megfelelő technológiára. Enélkül meg sem közelíthető a professzionális szoftverek megbízhatósága, minősége, kezelhetősége.

KŐFALUSI VIKTOR

Táguló μvilág

A gazdasági fellendülésre váró világban a számítógépgyártók barométere tartósan derült időt jelez.

A professzionális személyiszámítógép-előállítók minden eddiginél nagyobb választékot kínálnak. Az élenjáró cégek kereskedőinek nyilatkozatai jelentős eladásokat sejtetnek. Kérdés, hogy ez csak reklámcélokat szolgál-e, vagy újabb fellendülés kezdetét is jelzi. Amit azonban minden kétséget kizáróan megállapíthatunk, az az, hogy az alapvető építőelemekben, amelyeket a személyi számítógépek gyártásához felhasználnak, az egyre gyorsuló fejlesztés eredményeként jelentős változások tapasztalhatók.

Egerek és ablakok

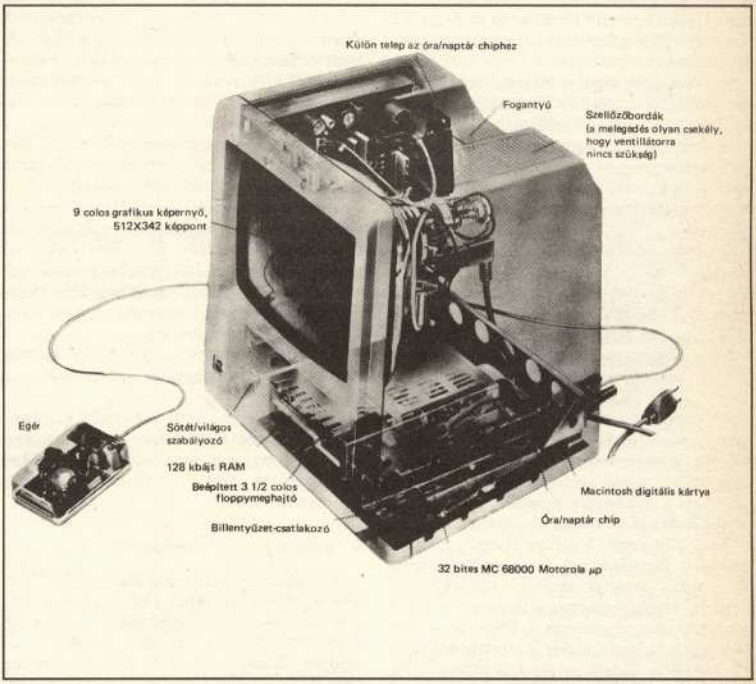
Az egyik változás az ember/gép kapcsolat bemeneti oldalának gyökeres átalakításában figyelhető meg. A fejlesztést gyaníthatóan egy korábbi elemzés eredménye ösztönözte. Az USA-ban 1982-ben hozták nyilvánosságra annak a statisztikai vizsgálatnak az eredményét, amely szerint a vezetők nagy része azért idegenkedik a számítógépek használatától, mert az ember/gép párbeszéd elsősorban a billentyűzet segítségével bonyolódik, és ehhez némi gépírói tudás szükséges. Ez a képesség pedig az esetek többségében nem tartozik a vezetők erősségei közé.

Ezért az ember/gép párbeszédet az Apple egy egér (MAUS) elnevezésű perifériával, a Hewlett–Packard pedig a kézérintésre érzékeny képernyő kifejlesztésével forradalmasította.

A két megoldás közül a drótfarkincával a géphez csatlakozó egér a népszerűbb; használata a hajdanvolt fényceruzára emlékeztet, de kezelése jóval kényelmesebb. Az egér egy golyós érzékelőt hordoz magában, amely mozgás közben elfordul, és ezzel adja a képernyőn megjelenő speciális jel számára a parancsot a mozgás irányára. Ha a jel megfelelő helyen áll, akkor az egéren levő nyomógombbal a kiválasztott hely paraméterei a számítógép tudomására hozhatók.

Az ember/gép kapcsolat kimeneti oldalán is kerestek és találtak alkalmas megoldást az „egértechnikához". Az új megjelenítési technika, amihez természetesen igen jó minőségű, nagy felbontású képernyők tartoznak, az ablak- (window) technika.

Korábban is alkalmaztak a fejlett szövegszerkesztő rendszerekben hasonló eljárásokat: a képernyőt több részre osztották, így a logikailag összefüggő, de különálló adatállományok egyidőben nyomon követhetők voltak. Az ablaktechnika ennek az eljárásnak tökéletesített változata, amelyet hardver- és szoftvertámogatással valósítanak meg. Ezzel a módszerrel újabb és újabb önálló képernyők határozhatók meg, és ezekben az adatokat egymástól függetlenül lehet megjeleníteni, mozgatni. Az eredeti képernyőtartalom az ablakkivágások mögött változatlan formában helyreállítódik az ablak „becsukása" után.

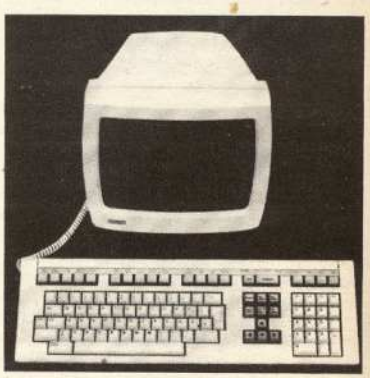


A Macintosh és az egere

Mikrokazetta és Winchester

A személyi számítógépek egyik gyengéje korábban az adattárolás szűkössége és bizonytalansága volt. A hajlékonylemezek (floppy) nem adtak megfelelő biztonságot, tárolókapacitást, és ráadásul igen érzékenyek is voltak. A személyi számítógépekbe most két technológiai újdonságot építenek be, ami nagymértékben megnöveli a gépek alkalmazási megbízhatóságát. Az egyik a nagy kapacitású Winchester technológiával működő beépített mágneslemez, amely 5–10 Mbájt adat tárolására képes, a másik pedig a mikrokazetta, amely – úgy tűnik – teljesen ki fogja szorítani a hajlékonylemez használatát.

Ma a mikrokazetták 5 $^{1}/_{4}$ és 3 $^{1}/_{2}$ colos kivitelben készülnek; ez utóbbi változatban is 1 Mbájt adat tárolására képesek. Az adathordozó kemény műanyag kazettában helyezkedik el. A lemez felületét csak az egységbe való behelyezéskor teszi szabaddá egy ablak. Így egy ma-



A Rainbow, azaz Szivárvány, mint munkahelyi számítógép

Az NSZK-ban 1983-ban értékesített professzionális személyi számítógépek

Gyártó	Darabszám	Részesedés %
1. Commodore	14 500	17,0
2. Apple	12 000	14,0
3. IBM	9 700	11,5
4. HP	7 100	8,5
5. Victor	5 850	7,0
6. Tandy	3 900	4,6
7. Triumph Adler	3 550	4,2
8. Olivetti	3 500	4,2
9. Taylorix	3 450	4,1
10. ITT	2 975	3,5
11. Egyéb	17 475	21,2

(Forrás: International Data Corporation)

A HP 150. Csak meg kell érinteni a képernyőt...

Isten hozott IBM – köszöntötte az Apple a PC-piacon csak 1983 elején megjelent versenytársát

1983-ban több mint két és fél millió Commodore mikroszámítógépet adtak el. A legújabb típus: a Commodore 8000.

roknyi kazettán akár közepes nagyságú vállalatok információrendszere is tárolható.

Irány: az IBM

Amióta a nagy cég beszállt a személyi számítógépek piacára, a verseny szinte egyesélyes lett. Ma minden gyártó az IBM személyi számítógépek vonalát követi, ehhez alkalmazkodik, ennek az árát kénytelen figyelembe venni. És az IBM minden területen támad. Létezik IBM–PC–junior, alap IBM–PC, hordozható IBM–PC, terminálemulációra alkalmas 3270–IBM–PC, nagy kapacitású IBM–PC–XT és a nagygépek szoftverjével (VM/CMS) működő IBM–PC–XT/370. Az IBM–PC-k ára az NSZK-ban 6-tól 25 ezer márkáig terjed, s nem könnyű a választás, különösen akkor nem, ha az összes többi gyártó is IBM-kompatibilis gépeket ajánl!

A Digital Rainbow-ja, a COMMODORE 8000-es sorozata, a Victor Sirius 1-e és még sok más cég gyártmányai is IBM–PC kompatibilisek. Árban vagy teljesítményben, avagy mindkettőben kedvezőbbet kell azonban nyújtani ahhoz, hogy a nagyétkű testvér mellett jusson valami a vásárlók tortájából.

Egy, csak egy legény van...

Az Apple az egyetlen, amely önállóan felveszi a versenyt az IBM-mel. Legújabb gépe, a Macintosh, minden újdonságot tartalmaz (32 bites M68000-es processzor, 128 k RAM, 64 k ROM, nagyfelbontású képernyő, egér, ablaktechnika, $3\,^1/_2$ colos mikrokazetta és az ára 7–8 ezer nyugatnémet márka). A terjesztést 1984. január 25-én kezdték meg, és azóta minden 27 másodpercben lejön egy „Mac" a futószalagról! Még ebben az évben 350 ezer darabot szeretnének értékesíteni, és ezzel az Apple legalább a második helyét kívánja megerősíteni a személyiszámítógép-gyártók világpiacán.

UNI(X)formizálódás

Csakúgy, mint a gyártásban, az alapszoftverellátásban is az IBM–PC-hez kell igazodni. Az MS/DOS mellett a Digital Research CP/M–86 és MP/M–86-os rendszerei tartják magukat, de abban a pillanatban, amint az IBM bejelentette, hogy az UNIX-ot (a Bell Laboratorium programterméke) az IBM–PC-re adaptálták, az összes gyártó kötelességszerűen az UNIX felé fordult. Így az alapszoftverek világa alighanem UNI(X)formizálódni fog.

És ahol a csata eldől

Az alkalmazási szoftverek piacán sokkal kisebb az előrelépés, mint ahogy azt a gyártók szeretnék. Bár minden jelentős cég igyekszik bizonyítani, hogy gépén százszámra vannak alkalmazási rendszerek, mégis ez képezi a szűk keresztmetszetet. A „Mac" szoftverellátására 500 új terméket kívánnak piacra dobni, de nyilvánvaló, hogy a hardver robbanásszerű fejlődését az emberi léptékkel haladó szoftverfejlesztés nem birja követni.

Így nem nehéz megjósolni, hogy az a cég fog győztesen kikerülni az értékesítési csatából, amely a felhasználói szoftverek területén is olyan átütő sikert tud majd felmutatni, mint a hardvertechnológiai újdonságok egybeépítésében.

FÖLDVÁRI

A BBC mikro-számítógép

A BBC mikro-számítógép

Valószínűleg nagyon kevesen ismerik itthon a BBC–Watford Electronics gépét, a BBC mikrót. Nem így van ez viszont Angliában, ahol ez az egyik legelterjedtebb személyi számítógép.

És erre meg is van minden alapja. A gép olyan előnyökkel és olyan kidolgozott szoftverrel rendelkezik, amit a COMMODORE és a Sinclair is méltán csodálhat.

Pillanatnyilag két változata van forgalomban: a 16 k-s MODEL A és a 32 k-s MODEL B. Vessünk egy pillantást a gépre! A BBC mikro klaviatúráján fellelhetjük a már elfelejtettnek hitt, a felhasználó által definiálható billentyűket. Ezek segítségével egy gomb lenyomásával több utasítás vagy parancs hajtható végre. A billentyűzet tartalmazza még az editáló funkció pozicionáló gombjait és a COPY másolóbillentyűt is.

A továbbiakban a MODEL B-t ismertetjük. A gép 7FFFH-ig programozható (RAM), felette található az elismerten gyors BASIC és a 6502 processzorhoz tartozó gépi kód fordítóprogramja. Ennek segítségével a BASIC programba – a programsorok közé – gépi nyelvű programokat is írhatunk, ami jelentősen megkönnyíti a 6502 assembler elsajátítását is.

A BBC mikro BASIC interpreterének számos előnyös vonása van. A hagyományos FOR...TO cikluson kívül az ALGOL-ból ismert REPEAT...UNTIL-t is magában foglalja. Procedúrák és függvények sokféleképpen definiálhatók. A fontosabb memóriaváltozókat és a belső valósidejű órát a BASIC-ből közvetlen függvényekkel érhetjük el (HIMEM, TIME). Jól használható a PAGE változófüggvény, amellyel kijelölhető a BASIC kezdőcím, és így a memóriában egyszerre egymástól függetlenül több programot is el lehet helyezni.

A legfőbb újdonság a javítás: a képernyő bármely részéről karakterek másolhatók az aktuális bemeneti sorba. Így rengeteg gépidő takarítható meg, hiszen mellőzhetjük ugyanannak a hosszabb kifejezésnek többszöri leírását.

A BBC számítógép hét színes képernyőmódban működhet. A MODE 7, az ún. TELETEXT mód a hagyományos ASCII grafikus karaktereivel rendelkezik; előnye az 1 k-s tárigény. A legjobb grafikus felbontás a MODE 1-ben érhető el. Ekkor 4 tetszőleges szín használható bármely pont kifestésére, és a képernyőn 1024 sor és 1248 oszlop helyezkedik el. A legtöbb szín – 8 villogó és 8 alapszín – a MODE 2-ben hozható elő; hátránya, hogy 20 k-t foglal le.

A grafika BASIC-ből is jól támogatott, például a DRAW és a PLOT utasításokkal. Az utóbbi segítségével nemcsak pontot, hanem szaggatott vonalat és háromszöget is rajzolhatunk. Sajnos körrajzoló utasítás nincs. A képernyőt külön videochip vezérli, amelyet a VDU utasítással is irányíthatunk.

A géphez számos periféria: nyomtató, hajlékonylemez, kazettás adatrögzítő, botkormány stb. illeszthető. A BBC mikrónak ADC (analóg-digitális konverter) bemenete, továbbá hanggenerátora is van, amelynek segítségével szintetizátor szinten szólaltathatók meg dallamok.

Megfontolandó, hogy nem lenne-e ott a helye a közoktatásban egy ilyen kitűnő BASIC-kel és élményszerű színes grafikával rendelkező gépnek?

TASSONYI KADOCSA

Rejtvény

Három nyolcbites tárolót nevezzünk el A-nak, B-nek és C-nek. A B-vel és C-vel külön-külön és együtt is végezhetünk műveleteket úgy, hogy a BC egy 16 bites tárolónak felel meg. Az A tartalma decimálisan 23, a BC tartalma 84. Ellenőrizzük a BC jobb oldali első bitjét. Ha ez 1, akkor adjuk a B-hez A-t, ha 0, akkor ne adjuk hozzá. Ezután toljuk jobbra a BC tartalmát egy bittel.

Mi lesz a BC tartalma decimálisan, ha az előbbi műveletet nyolcszor végezzük el?

A III. sz. modellező szakkör kísérleti receptjeiből I.

Végy egy írógépet, egy táblát krétával és Ló Pált, aki az osztály legjobb matematikusa és egyben legjobb tornásza is. Kösd össze őket (középre Palkó barátunkat), majd adj a kiváló matematikusnak néhány doboz gyufát és pár darab tejfölöspoharat (kimosva!).

Most jön a neheze: vegyél úgy tízezer borítékot, számozd meg őket egyesével, majd rakd sorba őket. Egyezz meg Ló úrral, hogy mit kívánsz tőle (számoljon a gyufákkal, rakosgassa őket egyik borítékból a másikba stb.), aztán abban is, hogy minden ilyen feladatnak legyen egy neve (mondjuk jelöld ezeket 1 és 255 közötti számokkal), a gyorsabb érthetőség kedvéért.

Magyarázd el barátunknak, hogy mi a gyufák funkciója: ezekkel lehet jelölni a számokat; mire kellenek a tejfölöspoharak: ezekkel lehet a műveleteket elvégezni (összeönteni, kivenni belőlük gyufákat); hogy a táblát és a krétát a veled való érintkezésre használhatja; végül a borítékok funkcióját is: ezekben lehet tárolni a gyufákat, ha a feladat ezt kívánja meg. Ha megkérdezi, miért kell jó tornásznak lennie, óvatosan add be neki, hogy gyorsan kell ide-oda tenni a gyufákat. (Nehogy elárulja, hogy másodpercenként több ezer feladatot kell elvégeznie!)

Ha mindezzel megvagy, ő is ért mindent, az írógépbe is tettél már papírt, elkezdheted kiváló számítógépedet használni. Gépelj a papírra – egymás alá – kódszámokat. Ezután kérd meg Pált, ugyan tegyen már a sorszámozott borítékokba (persze sorszám szerint) annyi gyufát, amennyi az éppen következő, feladatot jelölő kódszám volt. Ha kész, mutasd meg neki, melyik boríktól kezdje végrehajtani a feladatokat (ismét sorszám szerint!), majd ülj le. Most Ló barátunk nekiugrik, te üldögélsz. Később Pál (lógó nyelvvel) leáll, te izgatottan felugrasz, és megnézed, mi áll a táblán.

Ezzel véget ért első számítógép-modellezésed. Gratulálok! Te pedig gratulálj Ló Pálnak, aztán olvasd tovább a SZÁZLÁBÚT!

POHLY FERENC

A ZX80 programozása

A gépi utasításokat konkrét példákon fogjuk bemutatni. Az egyes utasításokat mindig elmagyarázzuk.

Elöljáróban néhány szó a bemutatott programok formájáról. A programot kétféle formában adjuk meg, ezek egymás mellett helyezkednek el. Az első oszlopban található a program szimbolikus kódolással, a második oszlop tartalmazza a gép számára közvetlenül értelmezhető tárgykódot. A szimbolikus kód könnyíti a programozó számára: könnyíti az egyes utasítások elsajátítását.

A futtatáshoz azonban a szimbolikusan írt programot le kell fordítani tárgykódra. A tárgykód egybájtos számokból áll, amelyeket be kell tölteni a memória szabad részére, például a POKE utasítás segítségével, majd rá kell adni a vezérlést CALL vagy USR utasítással.

Erről kicsit bővebben. A számítógépben mindig gépi program fut. A BASIC-ben megírt programot egy előre megírt gépi program értelmezi és végrehajtja. Ez az előre megírt gépi program az interpreter, amely állandóan a gép memóriájában található. Ha saját gépi programot futtatunk, akkor nem az interpreter fut, hanem közvetlenül a saját programunk. Megtakarítjuk tehát a BASIC értelmezésének idejét. Ezért sokkal gyorsabb a gépi program.

Lássuk a példaprogramot! Ez a program HT-n működik. Feladata a képernyő befehérítése. A program nem a lehető legegyszerűbb, de jól megvilágítja a gépi programozás elvét.

A program megértéséhez ismerni kell a képernyőn látható kép tárolásának módját. A HT-gépen erre a memória 3COOH–3FFFH című része szolgál. A képernyőn 1024 karakter fér el, a memóriában a látható karakterek ASCII kódjai találhatók. Először az első sor, aztán a második stb. sorjában. Minden karaktert egy bájt memória jut. A teljesen fehér képernyő esetén a videomemória 3COOH–3FFFH minden bájtjában a fehér karakter kódja a 191-es szám kell hogy legyen. Feladatunk tehát feltölteni a memóriát 191-gyel.

Lássuk a programot:

```
        LD,HL,3COOH     21 00 3C
        LD B,16         06 10
U2:     LD C,64         OE 40
U1:     LD (HL),191     36 BF
        INC HL          23
        DEC C           OD
        JR NZ,U1        20 FA
        DEC B           05
        JR NZ,U2        20 F5
        HALT            76
```

Az első utasítás az LD utasítás, jelentése tölts (load). Két argumentuma van, vesszővel elválasztva. Az utasítás a második argumentum értékét betölti az első argumentumba, ha az regiszter, vagy betölti az első argumentumban meghatározott memóriabájtra. Ez utóbbi esetre példa a negyedik utasítás. Ilyenkor a (HL) azt a memóriabájtot jelenti, amelynek címét a HL regiszterpár tartalmazza.

A kettősregiszterek feltöltésénél először az alacsonyabb helyiértékű, azután a magasabb helyiértékű bájtot kell megadni. Tehát a memóriába először a 21H számot kell betölteni, ez jelenti azt, hogy a HL regiszterpárnak akarunk értéket adni. Majd be kell adni ezt az értéket a megadott módon. Az első, kisebb helyiértékű bájtot az L regiszterbe, a második, nagyobb helyiértékű bájtot a H regiszterbe tölti a gép.

Az ötödik, INCrement utasítás eggyel növeli az argumentum értékét.

A hatodik, DECrement utasítás eggyel csökkenti az argumentum értékét.

A hetedik utasítás JR relatív ugrás. Ennek az utasításnak vagy egy, vagy két argumentuma van. Amennyiben egy, úgy feltétel nélküli vezérlésátadás történik. Ha két argumentum van, akkor az első a feltételt tartalmazza, amelynek teljesülése esetén megy végbe a vezérlésátadás.

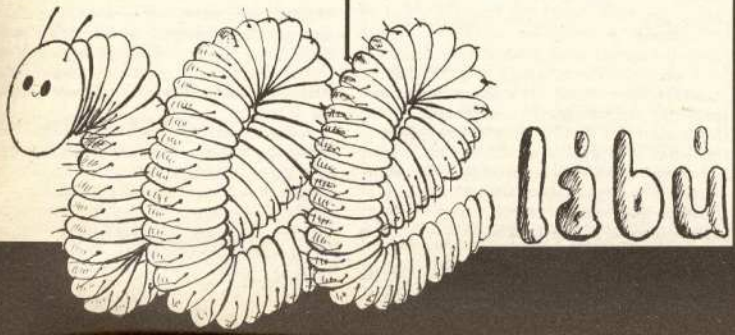
Jelen esetben akkor történik ugrás, ha a Zero flag nem nulla, vagyis ha az előző dekrementálás eredménye nem nulla. Ha azt akarnánk, hogy akkor történjen ugrás, amikor az eredmény nulla, akkor JR Z,U1-et kellett volna írni.

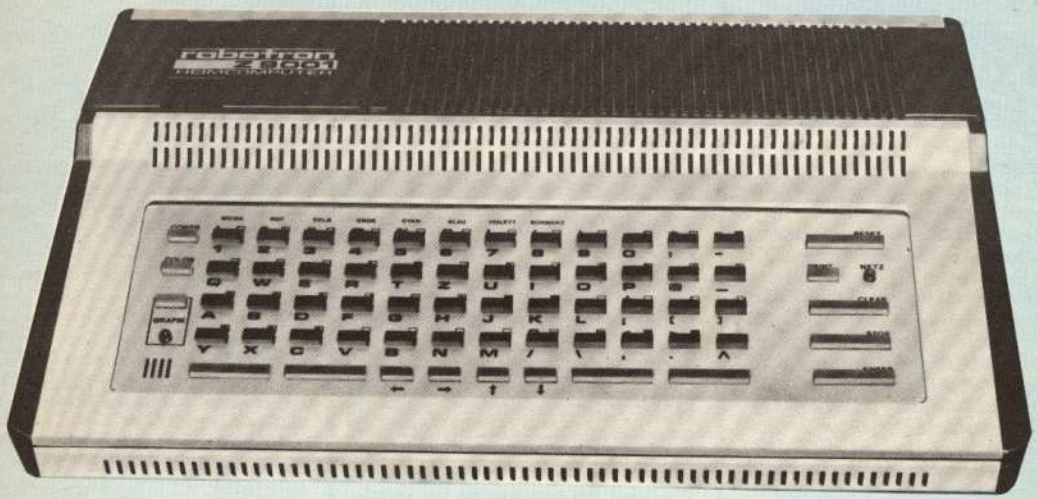
A tárgykódban a következőképpen kell kiszámítani azt, hogy hova ugorjon a futás. Mint tudjuk, a PC kettősregiszter mindig arra a címre mutat, ahol a következő utasítás van. Tehát a hetedik utasítás végrehajtásakor a PC már a nyolcadik utasításra mutat, vagyis a 05-re. Azt akarjuk, hogy a következő utasítás az U1 címkéjű utasítás legyen, tehát a PC a 36-ra mutasson. Ki kell vonni tehát a PC-ből hatot. Ezt úgy érjük el, hogy a JR NZ utasítás kódszáma után (20) odaírjuk azt a számot, amennyivel a PC-t változtatni akarjuk. Ez a szám egy bájt lehet. Vagyis 255-féle változtatást végezhetünk. Ha ide 1–129-ig írunk, akkor ezt az értéket a gép hozzáadja a PC-hez; ha 255–130-ig, akkor a gép 1-et, 2-t stb. levon a PC-ből. Az FA hexadecimális szám tehát mínusz hatot jelent.

Az utolsó utasítás megállítja a gép működését. Az újraindítás a RESET gomb segítségével történik.

A program működése a következő. Betöltjük a HL-be a videomemória kezdőcímét, 3COOH-t. B-be a sorok számát, a C regiszterbe az oszlopok számát. A HL-ben levő címre betöltjük a fehér karakter kódját, majd megnöveljük HL értékét. Kivonunk C-ből egyet, és ha C nem nulla, vagyis még van olyan oszlop, ahol nincs 191, akkor megismételjük az eddigieket. Majd hasonlóképpen ezt az egészet addig ismételjük, amíg B értéke nulla nem lesz, vagyis amíg az összes sorba be nem töltöttünk 191-et.

SIEBEN NÁNDOR





ILYEN A Z9001

Házi számítógép – Made in NDK

Az idei lipcsei Tavaszi Vásár egyik sztárja volt a Robotron Z 9001 házi számítógép. Jóllehet az NDK-ban mintegy két éve gyártják – elsősorban oktatási célra, a gépi kódú programozás gyakorlására – a Poly-Computer 880-at (Kombinat Polytechnik und Präzisionsgeräte, Karl-Marx-Stadt), az olcsó, nagyobb „tudású" új eszköz egy csapásra az érdeklődés középpontjába került.

A mintegy 5 kg súlyú, 30 × 40 × 8 cm méretű számítógép kezelésére 58, a számokat, betűket, jeleket és vezérlőgombokat tartalmazó klaviatúra szolgál. Háttértára közönséges kazettás magnetofon, 300 bit/s bitszinkronizált átvitellel, kijelzőként fekete-fehér vagy színes televízió használható. (A színes modult külön lehet megvásárolni, a vezérlőkártyát a szerviz építi be.) A képernyőn – választhatóan – 20 vagy 24 sorban 40 jel ábrázolható.

Az alapgép lelke egy U 880 D típusú mikroprocesszor. A tároló alapkapacitása 22 kbájt (ebből 16 kbájt RAM az adatoknak, a programoknak és a kazettáról betölthető BASIC interpreternek). A gép U 880 gépi kódban és BASIC nyelven programozható. A tároló 64 kbájtig bővíthető, 16 kbájtos RAM, illetve 12 kbájt kapacitású ROM modulokkal.

A programkazetták a számítógép működését oktató programot, BASIC interpretert, játékokat (logikai és reakciójátékokat, holdra szállást, Mastermind-ot stb.) tartalmaznak. Készült néhány oktatóprogram, valamint hobbival, otthoni tevékenységgel, illetve foglalkozással kapcsolatos alkalmazási program is (költségvetés, címjegyzék, határidőnapló, számlázás, modellvasút-vezérlés). Az üvegházi vezérlőprogram például a hőmérsékletet, a talajnedvességet, a páratartalmat figyelembe véve szabályozza a belső klímát (amihez természetesen utólag csatlakoztatott speciális be- és kimenet szükséges).

A gép alapkészletéhez tv-csatlakoztató kábel, két irányítókar (a játékokhoz), valamint az alapszoftver-csomag tartozik. Külön vásárolható kiegészítő berendezések: analóg-digitális átalakító, nyomtatóvezérlő, bővítő modul 16 ki- és bemeneti csatornára, további standard alkalmazási programokat tartalmazó ROM modul, valamint zenei kimenet (miniorgona).

A készülék előreláthatólag még ebben az évben forgalomba kerül. Tervezett, még jóvá nem hagyott NDK-beli ára magyar pénzben 10 ezer forint alatt van. Az első párezer darabból feltehetően csak kevés jut el az otthonokba, az iskolák – mint vásárlók – ugyanis előnyben részesülnek. Elvi megállapodás született a gyár és az illetékes minisztérium között, hogy ha igény lesz rá, további különféle oktatási programokat is készítenek.

A szakemberek elmondták, hogy az alapprogramokat a fogyasztók kívánsága szerint tudják bővíteni, de amíg a házi számítógép tömeges méretben nem kerül forgalomba, addig csak találgathatják a berendezéssel kapcsolatos igényeket.

Véleményük szerint korlátozza majd a felhasználók körét az a tény, hogy nincs a géphez elfogadható árú nyomtató. Úgy vélik, ha sikerülne egy körülbelül 500 márkás (kb. 2900 Ft) nyomtatót készíteni, akkor lehetővé válna, hogy például a kisüzemek könyvelésüket ezzel a géppel végezzék.

Megfelelő programmal szövegfeldolgozásra (adatbevitel és korrektúra) is alkalmas lehetne a Z 9001, ám mivel a felhasználónak többnyire a hagyományos értelemben vett kéziratra is szüksége van, ez az alkalmazás csak a távoli jövőben valósulhat meg. Bizonyos, Robotron által gyártott, elektronikus villanyírógépek (az S 6010-es vagy az S 6011-es) megfelelő interfésszel alkalmasak ugyan a nyomtató funkció ellátására, csakhogy ezek az írógépek viszonylag drágák.

A gép – a helyi telefonhálózat alkalmatlansága miatt – jelenlegi formájában nem használható se távadatfeldolgozásra, se két készülék telefonvonalon történő összekapcsolására. Ezért az ehhez szükséges elemeket be sem építették.

A hiányosságok ellenére a Robotron munkatársai remélik, hogy a Z 9001 az NDK határain belül és kívül egyaránt sikert arat.

RITTER P. TIBOR

Bulgária – mikroperspektívában

Az első bolgár elektronikus számítógép 1963-ban készült el, mágnesdobos operatív tárral, de sorozatgyártása nem indult be. Röviddel ezután jelentek meg a piacon az ELKA-6521, az ELKA-22 és az ELKA-25 típusú gépek.

A második generációhoz tartozó első számítógépet, a ZIT-155-öt 1968-ban a Szófiai Számítástechnikai Gyárban, a Fujitsu japán cég licence alapján állították elő.

1971-ben elkészült az első bolgár ESZR-termék: az ESZ-5052 típusú mágneslemezes tároló, 7,25 Mbájt kapacitással, majd 1972-ben az első processzor, az ESZ-2020. Ezt követte az ESZ-1022-es, 1980-ban pedig az ESZ-1035-ös rendszer. Az MSZR-ben Bulgária 1981-ben debütált az IZOT 1016 C (más néven SZM-4) gyártásbavételével.

Bulgáriát azonban a szocialista országokban elsősorban nem számítógépeiről, hanem mágneses tömegtárolóiról ismerik. A legtöbb, szocialista országban gyártott nagyszámítógép mellett bolgár szalag-, illetve lemezegység működik.

A mikroprocesszoros rendszerek területén Bulgária a Motorola 6800-as funkcionális megfelelőjét gyártja, SZM 601 néven. Mivel a többi szocialista ország ezzel szemben az Intel 8080 és a Z80 vonal mellett döntött, Bulgária egyre szorítóbbnak érzi az egyedüllétet (az óriási exportra számítva kialakított, hatalmas gyártókapacitások mellett exportja beszűkült, és magára maradt a szoftverfejlesztés- és adaptálás terén is). Ezért jelenleg éles viták dúlnak arról, hogy hogyan tovább a 16 bitesekkel: a Motorola 68000 vagy az Intel 8086 legyen-e a funkcionális analóg?

Motorola 6800 alapú mikroszámítógépet a szocialista országok közül Bulgárián kívül – a fő irányvonal mellett – csak hazánkban gyártanak (például HT 680X, N 68, OBT-16, MOTRA, SIMON-68). Ennek ellenére Bulgária igen aktívan próbálja „megdolgozni" a szocialista piacokat, például minden vásáron bőséges kínálattal vonul fel. Az eredmények még nincsenek arányban a befektetett energiával; hazánkban is csak néhány bolgár mikroszámítógép működik (többek között jelenlét-nyilvántartó rendszer).

Mikroszámítógépes rendszerek

Bulgáriában jelenleg elsősorban a professzionális mikroszámítógép-kategóriát gyártják. Széles választéktok hoztak létre ún. problémaorientált komplexumokból (POK). Ez azt jelenti, hogy a feladat funkcionális igényeinek megfelelően alakítják ki mind a hardvert, mind a szoftvert, és az így kifejlesztett típuskonfigurációkat forgalmazzák.

A POK-ok fontos jellemzője, hogy általában bolgárul beszélő, számítástechnikai szempontból laikus felhasználónak szánt, kulcsrakész alkalmazói rendszerek (jelenlét-nyilvántartó rendszer, benzinkutas rendszer, takarékpénztári rendszer stb.). Programfejlesztésre általában nem is alkalmasak (például a jelenlét-nyilvántartó rendszer kezelésére kizárólag numerikus és funkcionális billentyűzet szolgál). Ezek a korlátok egyrészt csökkentik az eszközárat, másrészt lehetőséget nyújtanak a kidolgozott alkalmazói célszoftverek többszöri értékesítésére (hardvervédelem a programmásolás ellen).

Nagy hátrányuk az SZM 601 alapú mikrogépes rendszereknek robusztus kivitelezésük: 160-200 kilót nyomnak asztalostul, a fiókba épített hajlékonylemezes tárolókkal, esetleg a beépített nyomtatóval együtt. A kivitelezés esztétikai szempontból is sok kívánnivalót hagyott maga után, míg végül 1983 tavaszán megjelent az új, formatervezett változat az IZOT 1024 és az IZOT 1025 képében. Azóta külföldre már csak ilyen formában szállítják a rendszereket.

Személyi számítógép

A bolgár mikroszámítógépek közül jelenleg csupán egyetlen típust gyártanak asztali kivitelben, Szófiában IMKO-2, Pravecben PRAVEC-82 néven. A gép Bulgáriában és a szocialista országokban készült alkatrészekből épül fel, és funkcionálisan megfelel az Apple II-nek. 64 kbájt operatív tára van, és háztartási kazettás magnetofonon kívül hajlékonylemezes tároló is csatlakoztatható hozzá. Ebben az esetben a gép lemezes operációs rendszer vezérlésével is működhet (azaz nemcsak a ROM-ban tárolt BASIC fordítót lehet hasznosítani), ami az adatállomány-kezelést is lehetővé teszi. Szintén lemezről hívható a PASCAL, FORTRAN és PILOT fordítóprogram. Újdonság az IMKO-2 terminálként való csatlakoztatása SZM-4 számítógéphez, ami gazdag grafikus lehetőségekkel jelentősen kibővíti az SZM-4 képességeit.

Az IMKO-2 egy éve a Neumann János Számítógéptudományi Társaság szervezésében nagy sikerű, országos bemutató-körúton mutatkozott be hazánkban. Különösen az iskolákban vált gyorsan népszerűvé ötletes, színekben tobzódó oktatóprogramjai miatt. A berendezésnek igen sok oktatóprogramja van, mivel Bulgáriában elsősorban iskolaszámítógépként forgalmazzák.

A többi szocialista ország döntésének megfelelően, 1984 tavaszán elkészült az U880D (Z80) alapú személyi számítógép prototípusa is. Az IZOT 1031 C típusú gépet a bolgár gyártmányú képernyős terminálok házában helyezték el, tárkapacitása 64 kbájt. Igen sokoldalú grafikai lehetőségei vannak. Különálló egységként két darab hajlékonylemezes tároló és nyomtató csatlakozik hozzá. Operációs rendszerként a CP/M 2.2-vel funkcionálisan azonos UMCO rendszert adják, annak megfelelő, széles körű szoftverellátással. A gép tömeggyártása az év második felében kezdődik, és a hírek szerint igen nagy sorozatra kívánnak berendezkedni.

Az IMKO-2 és az IZOT 1031 C kiváló tulajdonságaik következtében, reális áron könnyedén meghódíthatják a szocialista országokat.

Perifériagyártás

Bulgáriában két mátrixnyomtató-típust és egy margarétakerekes (30 karakter/s) mikronyomtatót gyártanak. Ezek közül kettő inkább a minigépekhez illik a maga 28–30 kiló súlyával, és csupán az IZOT 630 C mátrixnyomtató rendelkezik valóban mikroméretekkel: 2,5 kg súlyú, és max. 40 karakter szélességben nyomtat, A/4 méretű papírra.

Hajlékonylemezes tárolóból kétféle típust állítanak elő: az egyik 5 1/4" (mini), a másik 8" (normál) méretű. Az ESZ-5088 típusjelű kisebb tároló prototípusa a Shugart 400-nak, a nagyobbik ESZ-5074 típus pedig a Shugart 800-nak felel meg.

Sajnos mind a nyomtatókból, mind a hajlékonylemezes tárolókból az igények messze meghaladják a gyártott mennyiséget, így exportra igen kevés jut belőlük.

DR. BROCZKÓ PÉTER

Adok – veszek – cserélek

Ebben a rovatban rövid, szöveges, a mikroszámítógépekkel kapcsolatos hirdetéseket közlünk.

A díjszabás:

Közületeknek gépelt soronként (60 karakter) 100,– Ft, magánszemélyeknek az első sor 50,– Ft, minden további sor 20,– Ft. Az NJSZT tagjainak az első három sor ingyenes.

● **MEGVÉTELRE AJÁNLOM** operációkutatási programcsomagomat egyben és darabonként is. Tartalma: szállítási feladat, lineáris és kvadratikus programozás, König-feladat, RAS módszer stb. A programok VC-20, vagy C64 gépen futtathatók. Dévényi Zsolt, Miskolc-Egyetemváros E/2, 3515.

Szoftver termékek és számítástechnikai szolgáltatások kis- és középüzemek részére

Ismerje meg és vegye igénybe az OKISZ Szervezési és Számítástechnikai Vállalat mikroszámítógépes alkalmazói rendszereit

A VT 20/A és VT 30 mikroszámítógépen üzemelő alkalmazói rendszereink, illetve önállóan is bevezethető moduljai:

Rendelésnyilvántartás	– ügyfélnyilvántartás, – rendelések nyilvántartása, • rendelésszámonként, • ügyfelenként (vevői, szállítói), • cikkenként, • határidő szerint.
Készletgazdálkodás	– raktári analitika (forgalom), – raktári készletek kimutatása, – immobil készletek kimutatása, – ABC-analízis felhasználás szerint, – KSH anyagfelhasználási mérlegkészítés, – leltárfeldolgozás.
Anyagfelhasználás kimutatása	– a felhasználások analitikus forgalom szerinti nyilvántartása, – a felhasználások összesített kimutatása költségnemenként, költségviselőnként, egy adott időponttól kezdődően. **Az egységes rendszer további szolgáltatásai:** – szállítói számlalikvidáció, – anyagforgalom szintetikus gyűjtése főkönyvi számonként, raktáranként, mozgásirányonként, – bérmunka jellegű anyagforgalom kezelése, – használatban lévő fogyóeszközök nyilvántartása (szerszámkönyv), – érték szerinti anyagárkülönbözet-felosztás.
	További fejlesztési irányok:
– műszaki adatlapok kezelése	– darabjegyzék, műveletterv, erőforrás,
– termeléstervezés	– a termelési terv nyilvántartása, – bruttó anyagszükséglet meghatározása, – bruttó erőforrás-szükséglet meghatározása, – kapacitásterhelés egyensúlyvizsgálat, – beszerzési anyagszükséglet meghatározása,
– termeléselőkészítés	– anyagutalvány-készítés, – munkautalvány-készítés, – termeléskísérő dokumentáció készítése,
– termeléselszámoltatás	– termelés normatív elszámoltatása anyag és bér vonatkozásában.

A ROBOTRON 5100-as gépcsaládra kifejlesztett rendszerek:

1. Készletgazdálkodási rendszer
2. Főkönyvi könyvelési rendszer

További fejlesztési irányok:
– raktárirányítás,
– vevőrendelés,
– szállítói szerződések nyilvántartása,
– állóeszköz-nyilvántartás, normatívák,
– norma- és technológiai jegyzékek nyilvántartása,
– gyártmány- és gyártáselőkészítés,
– szükségletszámítás,
– termeléstervezés, kapacitásnyilvántartás,
– munkaügyi adatok nyilvántartása,
– béroptimum-számítás,
– adatrögzítés és előkészítés osztott feldolgozásoknál,
– on-line terminálként való felhasználás.

Az alkalmazási igényeknek megfelelő géptípus gazdaságos kiválasztását biztosítjuk. Új rendszereket fejlesztünk ki a VILATI FLOPPYMAT mikroszámítógépére és a bolgár gyártmányú IZOT 0220 M2 mikroszámítógépre.

Rendszereinkről részletes referenciát adunk. A megrendelők speciális igényeinek megfelelően kulcsrakészen adjuk át rendszereinket, a gépkezelők kiképzését, betanítását, a rendszer beindítását és a szervezési felügyeletet biztosítjuk.

OKISZ Szervezési és Számítástechnikai Vállalat Szervezési Főosztály Tel.: 426-936 Vállalkozási Iroda Tel.: 340–102 Levélcím: Budapest 8. pf. 247. 1445

Hazai szoftverkínálat mikroszámítógépekre 1984

Sokat beszélünk a mikroszámítógépekről, de elkövetjük azt a hibát, hogy egy kicsit eszközközpontúan, ami az új géptechnika megjelenésének tulajdonítható. Sőt, kezdetben nemcsak hogy az eszközre, hanem kifejezetten az alapgépre koncentráltunk, mit sem törődve a perifériák hiányával és árával. Hála a magyar fejlesztők, licencvásárlók és importőrök rugalmasságának, ma már a mikrogép árával összhangban lévő nyomtató beszerzése sem probléma.
Mi kell még? Műszaki kiszolgálás, oktatás, szoftverellátás. A műszaki kiszolgálást kikényszeríti az üzembe helyezett gépek rohamosan növekvő száma. Az oktatásról annyit, hogy a SZÁMALK (SZÁMOK) CP/M tanfolyamát terven felül prolongálni kellett.
Most helyezzük nagyító alá a hazai szoftverkínálatot. Alkalmat és alapot ad ehhez a KSH Számítástechnika-alkalmazási főosztályának az idei BNV-re összeállított katalógusa.

Földbe gyökeredzett az írógépem, amint az első számadatot le akartam írni. Ezért gyorsan megjegyzem: a következő elemzés sok adatot tartalmaz, de mi elsősorban az összefüggésekre, a tendenciákra koncentrálunk.

Közel négyszáz

Az 1983. évi szoftverkatalógus 8, mikrogépre készült szoftvertermékét ismertet. Az 1984. évi márciusi gyűjtés (ami nem terjed ki az oktatási és játékprogramokra) eredménye 381 szoftvertermék. Természetesen ez nem mind az elmúlt félévben keletkezett. Benne van ebben a gépkategória iránti növekvő bizalom és a piacra utaltság fokozódása is.

Milyen géptípusokra alkalmazhatók ezek a termékek? Némi összevonással 26 géptípushoz rendelhetjük a szoftvereket. Hogy az egyes gépek szoftverellátásának színvonalát megállapíthassuk, vessük össze a kidolgozott termékek számát az üzembe helyezett gépek számával. Ez utóbbi a rendkívül gyors változás miatt becslés, és nem ölelí fel az otthonokban használt gépeket (lásd *táblázat*).

Milyen feladatokat látnak el a programok?

Nézzük először a kérdést a gépi adatfeldolgozási termékosztályok szerint (lásd *ábra*).

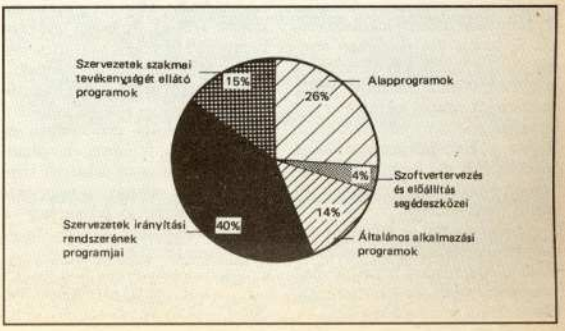
Ebből megállapíthatjuk, hogy – feltehetően a mikrogépek „laikus" alkalmazhatóságának követelménye miatt – a szoftverelőállítás segédeszközei csak nagyon kis hányadot képviselnek. Örvendetes viszont, hogy a szervezetek szakmai tevékenységét segítő programok aránya 15 százalék, ami azt jelenti, hogy ezekben az esetekben már a termelési folyamathoz kapcsolódik a mikroszámítógép.

Néhány jellegzetes alkalmazási terület

A mezőgazdaság tervezése – irányítása: 12 cég ajánlotta fel 23 szoftvertermékét, amelyek elsősorban az állattenyésztés termelésirányításának részfeladatait támogatják. Érdekes csomósodási pont a szállítás irányításával, elszámolásával kapcsolatos programtermékek. A katalógus e témakörben 11 különböző célú szoftvert kínál. És két kuriózum: nemzetközi vízügyi alkalmazás és galambsportlövészet.

Az alapprogramok köréből ki kell emelni a hálózati szoftvertermékeket. Megtalálhatók itt az ESZR és az MSZR gépekre kidolgozott terminálemulációk, a többfelhasználós mikrogépes operációs rendszerek és a különféle csatlakozási megoldások. Jelentőségük abban áll, hogy megteremtik az eltérő nagyságrendű gépkategóriák

Géptípus	Magyarországon forgalomban lévő szoftvertermékek száma	Az üzembe helyezett gépek száma (becslés 1984 végéig)
Tőkés import:		
ZX81 (min. 16 kbájt)		
Sinclair Spectrum	18	2000
LOGABAX	1	20
COMMODORE '64	98	4000
COMMODORE 8032	1	10
HP 98XX, 9515	6	200
Apple II	1	50
IBM-PC	2	30
Szocialista import:		
Robotron 1720/FD, 1750, 1840	11	600
Robotron 5110, 5120, 5130	21	300
Hazai termék:		
Aircomp 16	5	150
LABSYS 80	4	150
MOD 81, MOD 81 M	33	250
HT 680X	3	100
SYSTER	3	150
VARYTER	5	50
TAP 34	29	250
MO8X	50	600
PROPER 8	25	140
PROPER 16	15	250
FLOPPYMAT SP	28	400
VILATI UNIPROG	2	110
TPA-JANUS	9	30
TPA QUADRO	3	10
VPC, VPPC	3	50
VT 20	26	120
VT 20/A, /4	55	150
Valamennyi géptípusra	16	



kombinációinak lehetőségét, az adott feladat megoldásához legjobban illeszkedő kapacitásstruktúrát.

Egyéb szolgáltatások a szoftvertermékekhez

Ez az az eset, amikor alig hiszünk a szemünknek. A programok 90 százalékához jár dokumentáció, 82 százalékához tanácsadás, 73 százalékához garancia, 68 százalékához oktatás és kb. 40 százalékához installálás, adaptálás és követés. Ha ez szakmailag igényesen történik, akkor nem panaszkodhatunk.

Kik a készítők?

58 hazai intézet, vállalat, kisvállalkozás stb. jelentette be a termékjelleg követelményeinek megfelelő fejlesztéseit. A kidolgozók százalékos megoszlása:

Gyártó	15%
Profi számítástechnikai szolgáltató vállalat	32%
Funkcionális gazdasági egység (ahol nem főprofil a számítástechnika, például: bőrgyár)	5%
Kutatóintézet (beleértve az egyetemeket is)	13%
Kisvállalkozás	35%

Megítélésünk szerint a szoftvertermékeknek nem egészen 10 százaléka az adaptáció, a többi saját fejlesztés. Itt talán van tennivalónk. A kis mikrogépek tőkés relációból való behozatala mellett a hozzájuk készült eredeti szoftverek nagyobb arányú jelenléte is kívánatos lenne. Ha már a kidolgozóknál tartunk, érdekes megfigyelni a kereskedelmi funkció elkülönülését a mikrogépes szoftvertermékek terén: ezek 38 százalékát nem a kidolgozó forgalmazza. Nagygépes szoftvertermékeknél ez az arány sokkal inkább a kidolgozó felé húz.

Mennyibe kerülnek?

500 forint havi bérleti díjtól 600 ezer forint vételárig. Mivel rendkívül nagy a szórás, géptípusonként átlagot számítottunk, amiből megállapítható, hogy egy programtermék a hardver árának (tipikus konfiguráció) átlagosan 30 százaléka (kb. 10 és 50 százalék között van.) Ha figyelembe vesszük, hogy egy mikrogépen legalább 2-3 rendszer, alapszoftver, segédprogramok, esetleg programcsomagok futnak, úgy egy kitöltött rendszernél a szoftver ára megegyezik a hardver árával (nem számítva az egyedi alkalmazásfejlesztési, illetve karbantartási költségeket). Érdekes, hogy a szoftver/hardver árarány a tőkés adatokkal egyezik (itt nem szabad megfeledkeznünk a hazai magas hardverárszínvonalról). Ragadjunk ki három különböző árú géptípust, amelyekre viszonylag jó adataink vannak:

Géptípus	Hardverár	Szoftvertermékek ára	Szoftver átlagár	Szoftver átlagár/hardverár
COMMODORE 64	200 eFt	98	18 eFt	9,4%
TAP 34	420 eFt	29	120 eFt	29,0%
PROPER 16	750 eFt	12	80 eFt	11,0%

Az árak természetesen nemcsak a géptípussal, hanem a kidolgozóval is szoros korrelációban vannak. Vitathatatlan a kisvállalkozások alacsonyabb árszintje. Pedig az eladások számát tekintve az egy programtermékre jutó eladások száma a kisvállalkozóknál jóval alacsonyabb. Az összes programtermék közül harmincnak volt 10 feletti eladási száma.

A vétel mellett a termékek 10 százaléka esetében a bérletezésre is mód nyílik. A havi bérleti ár három megvizsgált terméknél 29, 35, illetve 38-ad része a vételárnak (a SZÁMALK OSAK gyakorlata szerint 1/24-e). A bérlet gyakran kedvezőbb pénzügyi konstrukció a felhasználónak, mint a vásárlás.

Befejezésül még egyszer hangsúlyozom, hogy az adatok csupán a márciusi gyűjtés eredményei, és mint minden felmérés, magában foglal egy bizonyos hibát. A vizsgált sokaság viszonylag nagy száma miatt mégis úgy érzem, hogy a hiba mértéke nem akkora, hogy túlságosan torzítaná a képet.

DR. ZÁRDA SAROLTA
SZÁMALK, Marketing főosztály

A VARYTER mikroszámítógép jellemzői

A VARYTER általános célú mikroszámítógép. Lehetőséget teremt kis- és közepes méretű, saját célú programok készítésére, fejlesztésére, futtatására. Alkalmas egyedi rendeltetésű, komplex feladatok számítástechnikai követelményeinek kielégítésére.

A mikroszámítógép felépítése

A VARYTER moduláris felépítésű. Egy egységet képez a képernyő, a billentyűzet, a tápegység és az elektronikai kártyákat tartalmazó kártyarekesz. A kártyarekeszben elhelyezett logikai áramköri kártyák kiválasztása az adott rendeltetésnek megfelelően optimálisan elvégezhető.

A mikroszámítógép fő egységei

A VARYTER 8 bites mikroprocesszor köré felépített mikroszámítógép. Perifériavezérlő készlete bővíthető.

Központi egység

A központi egység egy ZILOG Z80A típusú vagy egy Intel 8080 típusú mikroprocesszort tartalmaz. Memória címzési lehetőség 128 kbájt Ezen memóriatartományon belül 2 kbájtos bontásban az ÍRÓ/OLVASHATÓ és a CSAK OLVASHATÓ tár területe tetszőlegesen megválasztható. A memória kezelése 8 csatornás, közvetlen memóriahozzáférés egységén keresztül történik.

Képernyő

A tv-képcső mérete 31 cm. A normál tv képernyője a fáradékonyságmentes munkavégzés érdekében zöld előtéttel látható el. A képernyőn a karakterek mérete 12 × 6-os raszterpont. Az egyes sorok összeérnek, ami által lehetséges a félgrafikus megjelenítés. Egy sor 80 karakteres. A sorok száma 25. Egyidőben maximálisan 256 különböző karakter jeleníthető meg. Karaktergenerátora fix vagy tölthető lehet.

A képernyőn lehetséges grafikus megjelenítés is 256 × 256, vagy 512 × 512 pontmátrix mezőben. (Fejlesztés alatt.)

Billentyűzet

A VARYTER mikrogép billentyűzetének elrendezése és felépítése a SYSTER mikroszámítógép billentyűzetével egyezik.

Periféria-készlet

A VARYTER mikroszámítógép perifériái:
- hajlékony mágneslemezes tároló (maximálisan 16 db meghajtó egység; szimpla, dupla írássűrűség; egy-, kétoldalas hajlékony mágneslemezek)
- kompakt kazettás magnetofon
- nyomtató (mátrix, sor, margarétakerekes)
- lyukszalagolvasó, lyukasztó
- vonalkódolvasó
- speciális perifériák (jegy-, számlanyomtatók)
- plotter
- speciális mérőműszerek, adatgyűjtők, stb.

A perifériák illesztése a központi egység felé perifériavezérlők felhasználásával történik. Az általános perifériavezérlők a perifériához vagy soros vonalon (V24), vagy párhuzamos interfészen (BSI jellegű), vagy a gép belső buszán keresztül csatlakoztathatók.

A SYSTER, VARYTER mikroszámítógépek szoftvertámogatása

A SYSTER és VARYTER mikroszámítógépek működtetését saját önálló operációs rendszerek támogatják, de célrendszerekben lehetséges a feladatot kielégítő célszoftverek alkalmazása.

Z80-as mikroprocesszor esetén a működtető operációs rendszer a CP/M kompatibilis NETTY 2.4 operációs rendszer, 8080-as mikroprocesszor esetén pedig az MPOS operációs rendszer.

Általános használatú rutinok

Általános használatú rutinok gyűjteménye áll a felhasználó rendelkezésére. Ezek assembler programokból közvetlenül használhatók, magasabb szintű programokból a programnyelv elemein keresztül érhetők el.

SYSTER–VARYTER mikroszámítógépek MTA SZTAKI

Aritmetikai és konverziós rutinok
Aritmetikai és konverziós rutinok, matematikai függvények. Aritmetikai egységgel bővített processzor esetén hardver, egyébként szoftver végzi a számításokat.

Képernyő és klaviatúra-kezelő rutinok
Az ember-gép kapcsolat programozásához adnak jól használható eszközt.

Periféria-kezelő meghajtó rutinok
A különböző típusú perifériák egységes kezelését teszik lehetővé. A felhasználónak nem kell külön figyelmet fordítania a különböző hardverillesztő egységek speciális sajátosságaival.

File és könyvtárkezelő rutinok
A háttértároló kezelhetőségét és integritását ún. könyvtárosztályokra alapozott file-rendszer biztosítja (külön osztályba tartoznak pl. a forrásnyelvi, a futtatható formátumú, az általános szövegfile-ok, stb.). Az ide tartozó rendszerszolgáltatások: file-ok létrehozása, könyvtározása, megszüntetése; szekvenciális és random szervezésű file-ok írása és olvasása.

Debug rendszer
Assembler szintű felhasználói programok javításához és hibakereséshez ad hatékony támogatást.

Rendszerprogramok
A programfejlesztés során jelentkező feladatokra a rendszerkönyvtárba felvett rendszerprogramok használhatók.

Text Editor
Szövegfile-ok szerkesztésére és karbantartására készült program. A képernyő és a klaviatúra sajátosságait kihasználva egyszerűen kezelhető és szemléletes eszköz.

Assembler
A standard Intel és Zilog assembler nyelven írt programok fordítását teszi lehetővé. Az áttekinthetőséget és nyomkövetést speciális direktívák és a szegmentált fordítás lehetősége segítik.

A file-rendszer információs és karbantartó programja
Operátori parancsokkal van lehetőség a file-ok és könyvtárak tartalmának megjelenítésére, másolására, archiválására, diszkek tartalmának átrendezésére, stb.

Magasabb szintű programnyelvek
A SYSTER, VARYTER berendezéseken jelenleg rendelkezésre álló magasabb szintű programnyelvek
- BASIC
- FORTRAN
- PASCAL

Kommunikációs szoftver
Bonyolultabb alkalmazói rendszerekben alapvető követelmény a rendszerkomponensek között a megfelelő kommunikáció biztosítása. Az adott rendszer sajátosságai szerint eldöntendő, hogy mit kapcsolunk össze, hogyan és miért. A kommunikációs alapszoftvercsomag széles körű választékot kínál a megfelelő megoldás megválasztásához.

Az összeköttetés fizikai alapja
A területi elhelyezkedés és az adatforgalom mennyiségi és minőségi követelményei szerint választandó meg. A hardver és szoftverkészlet a következő lehetőségeket támogatja:
– Soros vonali interfésszel aszinkron vagy szinkron kapcsolat kis távolságokra közvetlen összeköttetéssel, nagyobb távolságokra bérelt vagy kapcsolt telefonvonalon vagy a nyilvános postai adathálózaton keresztül (V24/X. 21) kis- és közepes sebességgel, főleg terminál–gép kapcsolatra.
– Kis távolságra nagy sebességű párhuzamos összeköttetés (BSI) főleg gép–gép kapcsolatra.
– Kis- és közepes távolságokra lokális hálózatra kapcsolódás (pl. COAX kábel), nagy sebességgel, főleg elosztott rendszerek esetén.

Az összeköttetés módja
Egy fizikai kapcsolaton keresztül a kommunikáció alapját képező eljárások vezérlik az adatforgalmat. A szoftver a következő működési módokat teszi lehetővé:
– Egyszerű aszinkron terminálként közvetlenül gépre, vagy hálózati terminál koncentrátorra kapcsolódás.
– Szinkron terminálként számítógépre kapcsolódás az adott termináltípus emulációs programja segítségével (pl. BASIC vonali eljárással).
– X. 25 nyilvános postai csomagkapcsolt hálózatra kapcsolódás, mellyel a hálózatra kapcsolt bármely végberendezés elérhető.
– Lokális hálózatra kapcsolódásnál az adott hálózat protokollja szerint.

A felhasználás lehetőségei
A VARYTER egy kapcsolatban terminálként vagy számítógépként is szerepelhet. A következő alapszolgáltatások lehetségesek:
– Virtuális Terminál. A VARYTER úgy viselkedik, mintha az adott gép terminálja lenne.
– File Transzfer. A felhasználó file-okat másolhat a VARYTER és egy másik számítógép között.
– Program–program kommunikáció. A felhasználó összeköttetést teremthet a VARYTER-en futó programja és egy másik gépen azonos időben futó program között.
– Ez utóbbi kettőre alapozott ún. üzenetküldő szolgáltatás, mellyel a kapcsolatban álló felhasználók üzenetet küldhetnek bármely félnek.

Adatkezelő szoftver
A mikroszámítógépre egységes adatkezelő szoftver szükséges. Az adatkezelő rendszerek fejlettsége az egyszerű file-kezelő rendszertől az adatbáziskezelő rendszerig terjed.
A fejlesztendő mikroszámítógépes adatkezelő rendszer moduláris felépítésű. Egy-egy modul az adatbáziskezelés adott funkcióját látja el, de nem önálló adatbáziskezelő rendszer. Egy modul például megoldja BASIC nyelv kiegészítéseként a hatékony file-kezelést, másik a kezelhető adattípusok kiterjesztését (INTEGER-típusú változókon aritmetikai és logikai műveletek végezhetők), a harmadik a riport generálást, egy további az ellenőrzött űrlapkitöltésszerű adatbevitelt, stb. A kidolgozott modulokból lehetőség nyílik „zárt" adatkezelő rendszerek kialakítására, amely már sémageneráló alrendszert is tartalmaz. Ezen alrendszer jellemzője:
– az adattípusok „éles" definiálhatósága
– lehetőség az adatok közötti válogatás, a rendezett megjelenítés gyors végrehajtására
– biztonság az adatállomány véletlen elronthatósága ellen
– speciális eljárások: szövegszerkesztés, statisztikai feldolgozhatóság, riport generálás
– tranzakciószerű sémamódosíthatóság programozás nélkül.

A SYSTER, VARYTER mikroszámítógépes rendszerek forgalmazása

A SYSTER, VARYTER berendezések mikroszámítógépes rendszerekben történő forgalmazását a sárisápi Új Élet MGTSZ, az Elektromodul és az MTA SZTAKI közreműködésével létrehozott MICROKEY Kutatási-, Fejlesztési-, Termelési Társaság látja el.

Építsünk számítógépet! II.

Az első részben ismertettünk egy speciális billentyűzetillesztőt, amelyet számítógépünk billentyűzetének elkészítésénél fogunk használni (egyébként bármilyen más számítógéphez is használható, minden átalakítás nélkül). Felhívtuk az olvasók figyelmét arra, hogy az ismertetett megoldással kapcsolatos kérdéseikkel bizalommal megkereshetik a szerzőt. Az első cikk és további információk alapján két középiskolás diák hozzákezdett egy billentyűzet megépítéséhez. Az építéssel kapcsolatos elgondolásaikról, tapasztalataikról a későbbiekben beszámolnak.

Sorozatunknak ebben a részében elkezdjük a számítógép legfontosabb alkatrészeinek ismertetését. Elsőként a 6803 típusú integrált áramkörrel foglalkozunk.

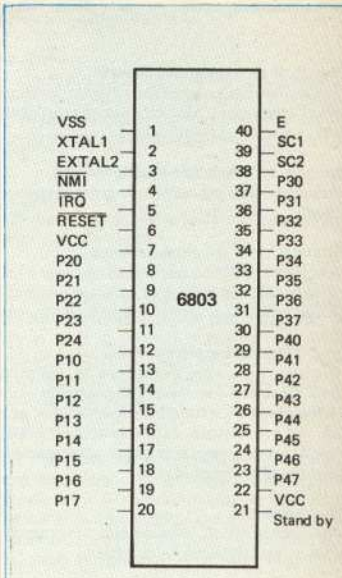
A 6803 a 6801 család egyik tagja. (A családról és az itt leírtakról részletesebb információk találhatók például a család adatlapjain.) A család tagjai a használó által programozható (és átprogramozható) üzemmódban képesek dolgozni. A lehetséges nyolcféle üzemmódból a 6803 hatfélét valósíthat meg: egychipes, kétféle vizsgáló, kiterjesztett nem-multiplexelt és kétféle kiterjesztett multiplexelt üzemmódot. Ezek közül számunkra csak az utóbbi fontos. A műveleti mód minden egyes RESET után újra beállítódik, így lehetőségünk van a működés közbeni módosításra is.

Az *1. ábrán* látható a 6803 lábkiosztása. Az 5 V tápfeszültségen kívül (7. láb) még egy tápfeszültség van (21. láb), amely az ún. „stand by" üzemmódban lehetővé teszi az információk tárolását mindössze 24 mAV teljesítményfelvétel mellett. Az órajel-frekvenciát párhuzamos rezgőkristállyal vagy TTL-szintű külső órajellel lehet biztosítani (2. és 3. láb). A rendszer busza számára ebből négyszeres leosztással állítja elő a chip az alapfrekvenciát (E jel a 40. lábon). Ez az órajel egy Schottky TTL IC-t képes hajtani. A különböző szintű megszakítások (RESET, NMI, IRQ) a 4.–6. lábakon keresztül közölhetők a chippel. A többi láb a négy „port".

A chip programszámlálója 64 kbájt címterülettel dolgozik. A tárterület első 32 bájtja a chip regiszterei számára foglalt. (A regiszterek tartalmát a *táblázat* ismerteti.) A következő 96 bájt a chipen kívüli terület, az ezt követő 128 bájt pedig belső, de szabadon programozható RAM. Ennek első 64 bájtja stand by állapotban is megőrzi tartalmát. A belső RAM az ún. RAM CONTROL REGISTER segítségével programozhatóan letiltható. Így ugyanazon címterületen dolgozhat a belső és külső RAM. (Mind a belső RAM-cím, mind a család ROM-memóriát tartalmazó változatainál a ROM-cím maszk-programozható.)

A chip belső rendszertechnikai felépítése látható a *2. ábrán*. A chip 6800 típusú (de valójában annak kiterjesztett utasításkészletű változata) mikroprocesszort, egy multiplexert, négy portot, egy soros kommunikációs illesztőt, egy timert és belső RAM-ot tartalmaz. A 6803 számos 6800 utasítást gyorsabban hajt végre azonos órajel-frekvencia esetén. Rendelkezik 13 további utasítással (ABX, ADDD, ASLD, BHS, BLO, BRN, LDD, LSRD, MUL, PSHX, PULX, STD, SUBD). Az A és B regisztereket mint D regisztert képes összefogni egyetlen két bájt széles regiszterként. A megszakításkezelés is kiterjesztett. Részben további megszakításfajták léteznek, részben kezelésük összetettebb.

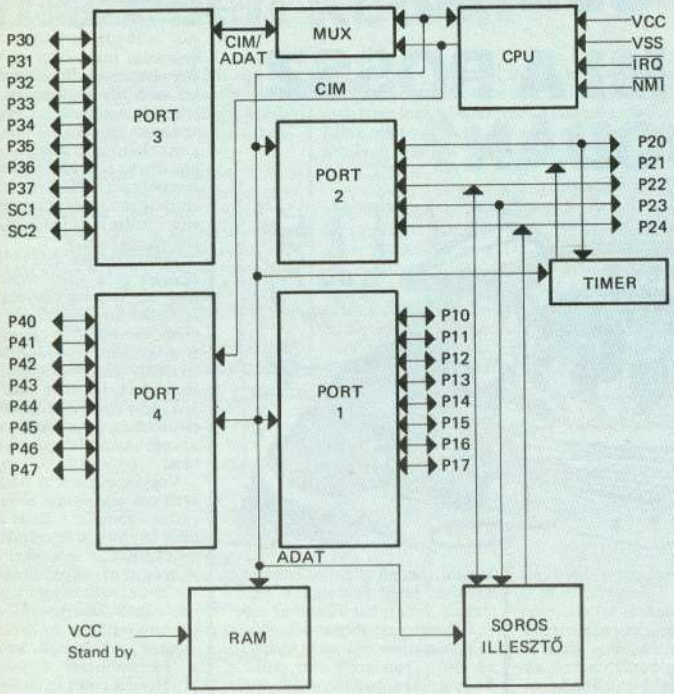
A timer rész tartalmaz egy 16 bites szabadon futó számlálót. Ennek használatával képes a bemenő impulzusok szélességének mérésére, különböző hosszúságú kimenő impulzusok elő-



1. ábra

Regiszter	CÍM	Megjegyzés
Port 1 Adatirány	00	1 = Kimenet, 0 = Bemenet
Port 2 Adatirány	01	mint 00
Port 1 Adattároló	02	
Port 2 Adattároló	03	
Port 3 Adatirány	04	mint 00
Port 4 Adatirány	05	mint 00
Port 3 Adattároló	06	
Port 4 Adattároló	07	
Timer Vezérlő és Állapotjelző	08	
Számláló Felső Byte	09	
Számláló Alsó Byte	0A	
Kimenet Összehasonlító Felső Byte	0B	
Kimenet Összehasonlító Alsó Byte	0C	
Bevett Bemenőjel Felső Byte	0D	
Bevett Bemenőjel Alsó Byte	0E	
Port 3 Vezérlő és Állapotjelző	0F	
Sebesség és Mód Vezérlő	10	
Adás/Vétel Vezérlő és Állapotjelző	11	
Vételi Adattároló	12	
Adási Adattároló	13	
RAM Vezérlő	14	
További fejlesztésre fenntartva	15–1F	

Bizonyos, általunk nem használt üzemmódoknál a 04–07 és a 0F regiszterek a használó számára nem hozzáférhetők.



2. ábra

állítására, időtúlcsordulás jelzésére és Baud-sebesség generálására különböző frekvenciákon. Mindezek a regiszterek használatával szabadon programozhatók.

A soros kommunikációs illesztő teljes duplex aszinkron kommunikációt tesz lehetővé. Kétféle soros adatformátumot és ötféle Baud-sebességet használhatunk. Az adatok vétele és küldése egymástól függetlenül, de azonos formátumban és sebességgel történhet. Itt is minden változat szabadon programozható.

A 6803 négy porttal rendelkezik. Ezek közül egyesek a műveleti módtól függő funkciókat látnak el. Minden port szabadon beállítható mint be- vagy mint kimenet. A Port 2 alsó három bitje RESET után beolvassa a műveleti mód kódját, és a 6803 ennek megfelelő üzemmódban dolgozik tovább. Ezt követően a port bemenetként programozódik. Öt bitje egyenként programozható, vagy alsó két bitje a timer be- és kimeneteként, a további három bit a soros illesztő adat be- és kimeneteként, illetve órajelként.

A Port 3 az általunk használt üzemmódban mint multiplexelt cím- és adatport működik. A hozzá tartozó két vezérlőjel közül az egyik mint címzésvezérlő (AS), a másik mint írás-olvasás vezérlő (R/$\overline{W}$) szerepel. A Port 4 a felső nyolc cím kiadására szolgál abban az üzemmódban, amelyet használni fogunk. A Port 1 szabadon programozható párhuzamos be/kimenő port.

Amennyiben a cikkben szereplő fogalmak, megoldások az olvasók számára ismeretlenek lennének, úgy ezekkel kapcsolatban is keressék meg a szerzőt. DR. SIMONYI ENDRE

Kabalapályázatunkra érkezett

3.

Sárközi Antoniette, Budapest

MI A RÉGI AZ ÚJBAN ÉS MI A ROSSZ A JÓBAN?

A napokban egy zömmel fiatalokból álló asztaltársaság körében a számítástechnikáról, mikroelektronikáról lángolt fel a vita. Hamarosan kellemetlen helyzetbe kerültem: nos, kinek van inkább igaza? Felkértek, hogy én, az idős ember, aki nemcsak koromnál fogva, hanem szakmámat tekintve is távol állok a tárgytól, de azért eleget hallhattam, olvashattam róla, mondanám el véleményemet. Elmondtam, most le is írom: túl a hetvenen hogyan képzeli egy „öreg vasas" ennek a technikának a távlatait.

A múltba nézést használom fel a jövőbe tekintéshez, mert ez az egyetlen biztos pont az elinduláshoz.

Születésem táján (1910) már eltűnőben voltak Gárdonyi Géza figurái, a Göre Gábor, a Durbints ságor és a Kátsa cigány, de az autó megjelenése kis városunkban még csodának tűnt. Kaposvár utcáján először a Vöröskereszt tér felől hallottuk a dübörgő hangot; a város apraja-nagyja kirohant az utcára. Egy négykerekű, ló nélküli tünemény száguldott el előttünk. Emberek ültek benne. „Ótomobil" – mondták a tudákosok. Aztán még egy-két ilyen csoda, és Kaposvárt csak mi, gyerekek szaladtunk ki az utcára, ha autót hallottunk közeledni.

A háború évei alatt már a repülőgép sem tűnt olyan különösnek; hírből tudtunk róla. Hat-hét éves koromban mindennap kimentem a közeli repülőtérre, nézegetni a felle szálló, gyenge kis gépeket. Lezuhanni is láttam egyet.

A húszas években a rádióval lepett meg bennünket a tudomány. És amikor a Telefongyárba kerültem inasgyereknek, ott hallottam a gyár mérnökéről, Mihály Dénesről, aki a távolbalátás technikájával foglalkozott, de akkor már Angliában élt.

Nos, ha ez is meglesz, mi kell még; van még valamire szükség? Úgy véltük, aligha. A második világháború, mint lényegében újat, az atomfizika felhasználását hozta a minél több ember és minél nagyobb vagyon elpusztítására. De akkor már sejtettük, hogy az atomfizika az emberiség javára is szolgálhat.

Hát így edződtünk évtizedeken át a csodák befogadására. Konyhaszekrényemen és csuklómon ott a kvarcóra, mutatja a pontos időt. Működését nem értem, de könnyedén tudomásul veszem létét, nem csodálkozom. Azt hiszem, a kevés hozzáértőt leszámítva, így van ezzel mindenki.

De míg idáig jutottunk, a technika fejlődésén kívül más is történt. Politikai szempontból már régóta kettészakadt a világ. A kapitalista rendszer ma még gazdaságilag erősebb, iparilag fejlettebb, mint a szocialista; és nem is biztos, hogy ez csak hátrányt jelent számunkra.

Múltba tekintésemben eddig eljutva, már elég bátor vagyok ahhoz, hogy az újabb csodatechnikának, amely a múlt technikáiból nőtt ki, és már nem is csoda, a jövőjéről beszéljek. A kapitalista ipar fejlettsége, a piacokért, a profitért folyó kíméletlen harc az egymással való versengésben arra kényszeríti a tőkés vállalkozásokat, hogy mindig újabb és újabb ipari kacatokkal árasszák el a világot.

A számítás- és robottechnikában még a fejlett kapitalista államok is irigyen követik Japánt, ahol már a robotgépek csápjai közül kerülnek ki a „meseautók". A piacok egymásra hatása következtében és természetesen a gazdasági haszon reményében a szocialista államok is egyre nagyobb mértékben alkalmazzák ezt a legfejlettebb technikát. El vagyunk ugyan maradva az alkalmazásban, de nem ez a legfontosabb problémánk jelenleg, hanem rendszerünk biztonságának, gazdaságunk stabilitásának megőrzése.

A lényeges különbséget én nem abban látom, hogy a két társadalmi rendszer melyikében alkalmazzák a legtöbb számítógépet, a legtöbb robotot, hanem abban, hogy a kapitalizmusban ezzel feleslegessé válnak és utcára kerülnek a folyton többet követelő, a jobb sorsukért sztrájkoló munkások. Hogy a robotokat is munkások készítik? Ugyan, hiszen pénz és befektetés kérdése, hogy önmagukat reprodukáló robotokat állítsanak elő.

Véleményem szerint a szocialista rendszerben erre az új technikára elsősorban azért van szükség, hogy az alkalmazásával feleslegessé vált munkaerőt a népgazdaság más területeire irányíthassák. Ebben a kétféle felfogásban fejlődik a két rendszer számítás-, irányítás- és robottechnikája. Ez van ma, és bárhogyan is van, jól vagy rosszul, én másképpen szeretném látni ennek a technikának a jövőjét.

Vannak egyrészt a fejlett ipari államok, ahol sokan dúskálnak a javakban. Másrészt pedig vannak a fejletlen államok, amelyeknek lakossága az egész Föld lakosságának 70–80 százalékát is eléri, és ahol évente milliók pusztulnak el alultápláltság és a nyomor által terjesztett betegségek következtében. Hiába marasztaljuk el a malthusiánusokat; a tények mintha őket igazolnák.

A Föld népessége gyorsabban növekszik, mint a megtermelt élelmiszerek mennyisége. Ezen a téren nem képes az emberiség utolérni magát. És ez mindaddig így lesz, míg az ipari kapacitást csak azért növelik, hogy még nagyobb tömegben állíthassák elő a szórakozást, a túlzott kényelmet biztosító javakat. De ez még a kisebbik baj; a nagyobb az, hogy a haditechnika mechanizmusa is arra az elgondolásra épül, hogy a számítás- és a robottechnika segítségével pusztítsák el az ellenfelet.

Ez a fenyegetés a fejünk fölött marad mindaddig, míg győz a józan ész. Én nagyon bízom abban, hogy a szocialista útra lépő népek valamilyen módon – de mindenképpen háború nélkül – jótékonyan hatnak a világ másik felére. Ekkor jön el a robotok igazi, valóban az egész emberiség javára szolgáló működésének ideje. Fel lehet használni majd őket olyan munkák elvégzésére, amelyekre az ember nem képes. Annyira tökéletesek persze sohase legyenek, hogy az ember kezéből teljesen kivegyék a munkát, mert anélkül már nem lesz élet az élet. Mindig kell egy kis verejtékezés. Földi paradicsomban élni nagyon rossz és unalmas lehet; erre igazán jó példa a paradicsomi eredeti legendája.

Természetesen van olyan elképzelésem is a jövő technikájáról, hogy az emberek kényelmét, egyéniségük kibontakozását fogja szolgálni, megkönnyíti egymással való érintkezésüket, személyes kapcsolataikat, kisebb-nagyobb ügyeik elintézését, lehetővé teszi az újságok, folyóiratok, könyvek újfajta előállítását, terjesztését – általában az egész nyomdai technika átalakítását.

Vegyük elsőnek a telefont. Nekem van telefonom, több százezer ember számára viszont a telefon csak óhaj. Ha a jelenlegi módszerrel akarjuk az igényeket kielégíteni, még az ezredfordulóig sem sikerül, és akkorra ismét újabb százezrek állnak elő igényléseikkel. Ez így nem tartható. Az örökös kábelgondok, a beázások, szakadások, a karbantartási és munkaerőproblémák miatt az én szememben ez az egész elavultnak tetszik. Más utat kell járnunk. Talán nem lehetetlenség az az elképzelésem, hogy például a jelenleg már létező rádiótelefont kellene olyan irányban fejleszteni, hogy a ma telefonközpontjai adó-vevő központokként létesítsenek kapcsolatot az előfizetők között, esetleg telefonvíziós rendszerben. Ez ma már csak pénz kérdése lehet.

Megszabadulhat az emberiség a mérhetetlen mennyiségű papírtömeg gyártásától és felhasználásától. Újságra, folyóiratra, könyvre előfizethetnénk úgy, mint eddig, a papírra azonban nincs szükség; kapok helyette mikrofilmet, a tartalmát pedig levetítem a képernyőmön. Vagy közvetlenül egy nyilvános információs hálózatra kapcsolódom.

Lehetne még tovább folytatni: minden bizonnyal sorra kerül olyan eszközök alkalmazása is, amelyekről én még csak nem is álmodom. Legfontosabb azonban az, hogy ezeket ne egymás boszszantására, hanem örömére használjuk. Ha most lennék 20 éves, talán meg is érném, hiszen nincs közöttük egy sem, amelynek elemei ne volnának már ma meg. Csak technikus és pénz kell az ügyes összekombinálásukhoz.

JAKAB GYÖRGY

Ismeretlen kódok

A nyolcbites mikroprocesszorok ún. egy bájt hosszúságú utasításai 256-félék lehetnek. A kereskedelmi forgalomban használt leggyakoribb típusoknál a gyártók által közölt utasításkészlet azonban ennél kisebb.

Most induló sorozatunkban ismertetjük azokat az ún. nem deklarált kódokat, amelyekre az adott mikroprocesszor „értelmes" választ ad. Megjegyezzük, hogy ezek a kódok az adott mikroprocesszor egyes sorozatainál esetleg eltérő eredményt adhatnak. Az olvasók ezzel kapcsolatos tapasztalatait közölni fogjuk. Közöljük az olyan tapasztalatokat is, amelyeket más kódokkal kapcsolatban szereztek.

Először a MOS Technology 6502 típusú mikroprocesszorával foglalkozunk. Ez a mikroprocesszor-típus – a hazai irodalomban megjelent közlésekkel ellentétben – a mikroszámítógépgyártásra legnagyobb példányszámban felhasznált nyolcbites mikroprocesszor (ezt használták az olyan „milliós" gépeknél, mint az Apple II, a VIC–20, C–64).

A táblázatban közölt kódok közül hiányoznak mindazok, amelyeknek alsó „fél bájt" értéke (másként nibble) 7 vagy F hexadecimális szám. Mivel $7_{16} = 0111_2$, $F_{16} = 1111_2$, azaz mindkettőnél közös az, hogy a legalsó három bit egyes, és ez más kódoknál nem fordul elő, ezért a csoportba tartozik minden olyan kód, amelynek a legalsó három bitje egyes.

Ø7-kód. Balra lépteti az utasítást követő bájt által megcímzett „zero page"-hely tartalmát, majd az eredményt VAGY-kapcsolatba hozza az akkumulátorral, és az akkumulátorba teszi. Összehasonlítva a Ø6 és Ø5 utasításkódokkal, azt találjuk, hogy ez az új utasítás helyettesíti azt a kettőt, és így 4 bájt helyett 2 bájt is elég.

ØF-kód. Azonos az előzővel, csak 2 bájt által meghatározott, ún. abszolút címet használ.

17-kód. Azonos az elsővel, csak a cím az X-regiszter által kijelölt, ún. indexelt címzésmódú.

1F-kód. Azonos az előzővel, csak a cím abszolút indexelt.

27-kód. Hasonló az elsőhöz, csak léptetés helyett forgatás, VAGY-kapcsolás helyett ÉS van. A 2F, 37, 3F a hex 2Ø értékkel alacsonyabb, ugyanilyen mértékben módosított párja.

47-kód. Hasonló az elsőhöz, csak jobbra léptet, VAGY-kapcsolás helyett KIZÁRÓ VAGY van. A 4F, 57, 5F a hex 4Ø értékkel alacsonyabb, ugyanilyen mértékben módosított párja.

67-kód. Hasonló az elsőhöz, csak léptetés helyett jobbra forgatás van és VAGY-kapcsolás helyett összeadás az ún. CARRY-bit figyelembevételével. A 6F, 77, 7F a hex 6Ø értékkel alacsonyabb, ugyanilyen mértékben módosított párja.

87-kód. Az akkumulátor és az X-regiszter tartalmát ÉS-kapcsolatba hozza, majd az eredményt az utasítást követő bájt által kijelölt „zero page"-címre teszi. A 8F hasonló, csak a cím abszolút. A 97 is hasonló, csak a cím indexelt és az indexelés az *Y-regiszter* szerinti.

9F-kód. Az akkumulátor tartalmát hex 4 értékkel ÉS-kapcsolatba hozza, és az eredményt az akkumulátorba teszi.

A7-kód. Az utasítást követő bájt által kijelölt „zero page"-cím tartalmát betölti mind az akkumulátorba, mind az X-regiszterbe. Az AF-kód hatása azonos, csak a cím abszolút. A B7-kód hatása is azonos, csak a cím indexelt és az indexelés az Y-regiszter szerinti. A BF-kód hatása szintén azonos, csak ez is indexelt, és az indexelés az Y-regiszter szerinti, de a cím abszolút, nem „zero page".

C7-kód. Eggyel csökkenti az utasítás utáni bájt által kijelölt „zero page"-cím tartalmát, majd az eredményt összehasonlítja az akkumulátor tartalmával, és az akkumulátorba teszi. A CF-kód hatása azonos, csak a cím abszolút. A D7-kód hatása is azonos, csak a „zero page"-cím indexelt az X-regiszterrel. A DF-kód hatása is azonos, csak az abszolút cím az X-regiszterrel indexelt.

E7-kód. A hatás hasonló az előzőhöz, csak növelés csökkenés helyett, összehasonlítás helyett a CARRY-bitet is figyelembe vevő kivonás van. Az EF, F7, FF kódok a hex 2Ø értékkel alacsonyabb, ugyanilyen mértékben módosított kódok párjai.

Az eddig felsoroltak mind egy csoportba tartoztak. Az utolsó 9E-kód nem azonos kódtípusú. Ennek hatása hasonló a 9F-kódéhoz, csak a kapcsolás az X-regiszterrel történik.

A kódok kényelmesebb használata érdekében a szokásos táblázatos formában is összefoglaljuk őket. A *táblázat* jelölésmódja is a szokásos. Reméljük, hogy ezek a teljesítőképes utasítások elősegítik jobb programok írását.

S.E.

NÉV	MŰVELET	FORMA	KÓD	CIKLUS-SZÁM	BÁJT-SZÁM	NZCV–BIT HELYZETE
ANDX	A ÉS X→M	ANDX$aa	87	3	2	– – – –
		ANDX$aa,Y	97	4	2	– – – –
		ANDX$aaaa	8F	4	3	– – – –
DCMP	A–(DECM)	DCMP$aa	C7	8	2	+ + + –
		DCMP$aa,X	D7	1Ø	2	+ + + –
		DCMP$aaaa	CF	1Ø	3	+ + + –
		DCMP$aaaa,X	DF	11	3	+ + + –
ISBC	A–(INCM)–C→	ISBC$aa	E7	8	2	+ + + +
	A,C	ISBC$aa,X	F7	1Ø	2	+ + + +
		ISBC$aaaa	EF	1Ø	3	+ + + +
		ISBC$aaaa,X	FF	11	3	+ + + +
LDAX	M→A,M→X	LDAX$aa	A7	3	2	+ + – –
		LDAX$aa,Y	B7	4	2	+ + – –
		LDAX$aaaa	AF	4	3	+ + – –
		LDAX$aaaa,Y	BF	4	3	+ + – –
RLAN	(ROLM) ÉS	RLAN$aa	27	8	2	+ + + –
	A→A	RLAN$aa,X	37	1Ø	2	+ + + –
		RLAN$aaaa	2F	1Ø	3	+ + + –
		RLAN$aaaa,X	3F	11	3	+ + + –
RRAD	(RORM)+A+	RRAD$aa	67	8	2	+ + + +
	+C→A,C	RRAD$aa,X	77	1Ø	2	+ + + +
		RRAD$aaaa	6F	1Ø	3	+ + + +
		RRAD$aaaa,X	7F	11	3	+ + + +
SLOR	(ASLM) VAGY	SLOR$aa	Ø7	8	2	+ + + –
	A→A	SLOR$aa,X	17	1Ø	2	+ + + –
		SLOR$aaaa	ØF	1Ø	3	+ + + –
		SLOR$aaaa,X	1F	11	3	+ + + –
SREO	(LSRM) KIZÁ-	SREO$aa	47	8	2	+ + + –
	RÓ VAGY	SREO$aa,X	57	1Ø	2	+ + + –
	A→A	SREO$aaaa	4F	1Ø	3	+ + + –
		SREO$aaaa,X	5F	11	3	+ + + –
TSTA	A ÉS # $Ø4→A	TSTA$aaaa	9F	4	3	– – – –
TSTX	X ÉS # $Ø4→A	TSTX$aaaa	9E	4	3	– – – –

Kedves Olvasó!

Reméljük, hogy a rovatnak nemcsak olvasói és a feladatoknak szorgalmas megoldói, hanem írói is lesznek.

A feladatokat – melyek megoldását mindig a következő számban ismertetjük – különböző korosztályok számára írjuk ki. A csoportok megkülönböztetésére római számokat használunk, a következőképpen:

I. 8–14 évesek
II. Középiskolások és szakmunkástanulók
III. Felsőfokú oktatási intézmények hallgatói
IV. Dolgozók

A megfejtéseket a rovat szerkesztőjének nevére küldjék:
Neumann János
Számítógéptudományi Társaság
Garádi János
Budapest V., Báthori u. 16.

Minden megfejtést jól olvashatóan, külön lapon kérünk. Minden lapon szerepeljen a feladat száma és a megoldó neve. A borítékban a neve és címe mellett mindenki tüntesse fel korosztályát és munkahelyét (iskoláját).

A legjobb, legötletesebb megoldásokat beküldőjük nevével együtt közöljük.

A kitűzésre szánt feladatokat és megoldásukat szintén a fenti címre várjuk.

Az alábbi feladatok beküldési határideje 1984. szeptember 15.

Feladatok

1.

Készítsünk nyugdíjjárulék összegét meghatározó programot. Az 1984. január 1-től érvényes nyugdíjjárulékok mértéke:

3%	–2100 Ft-ig
4%	2101–2600 Ft
5%	2601–3300 Ft
6%	3301–4300 Ft
7%	4301–5300 Ft
8%	5301–6300 Ft
9%	6301–7300 Ft
11%	7301–8300 Ft
12%	8301–10 300 Ft
13%	10 301–12 300 Ft
14%	12 301–14 300 Ft
15%	14 301– Ft-tól

A nyugdíjjárulék összegének meghatározásánál az általános kerekítési szabályt kell alkalmazni. (0,50 Ft és az ezt meghaladó értékek egész forintnak számítanak.)

Beküldendő a személyi számítógépre készített magas szintű programnyelven elkészített program.

(Ajánlott a II–IV. korosztály részére.)

2.

Készítsünk programot, amely megmondja, hogy a háromjegyű egész számok közül melyiknek van a legtöbb osztója és melyek azok. Ha több szám is eleget tesz az előbbi feltételnek, akkor közöljük azok értékeit is.

Beküldendő a BASIC program, az egész számok értékei, valamint osztóik értékei és száma.

(Ajánlott az I–IV. korosztály részére.)

3.

Határozzuk meg a $\frac{K}{M}$-nek tizedestört alakjában a tizedesvessző utáni L-edik számjegyének értékét anélkül, hogy az előző jegyeket mind kiszámítanánk.

A program bemenő adatai: K, M, L.

A program az alábbi értékeket közli:

- Végtelen tizedestört esetén a kérdéses számjegy értéke;
- Véges tizedestört esetén ha van L-edik számjegy, különben tetszőleges üzenet.

Beküldendő a feladat blokkdiagramja és a BASIC, PASCAL, FORTRAN nyelven megírt program.

(Ajánlott az I–II. korosztály részére.)

4.

A rendelkezésünkre álló számítógépen a véletlenszám-generátor felhasználásával készítsük el az alábbi programot.

A program kiír a képernyőre két egész számot (max. 3 jegyű), melyet a gép véletlenszerűen ad meg. A gép előtt ülő felhasználó pedig gondol szintén egy egész számot, majd a gép által kiírt két számot ezzel a számmal elosztja, és közli a géppel az első és a második szám esetében a maradékok értékeit. Ezt követően a program kiszámítja az osztó értékét és a képernyőre kiírja.

Beküldendő a BASIC, PASCAL, FORTRAN, ALGOL nyelvek közül valamelyiken megírt program listája és rövid használati útmutatója.

(Ajánlott I–IV. korosztály részére.)

Megoldások az előző számban közölt feladatokhoz

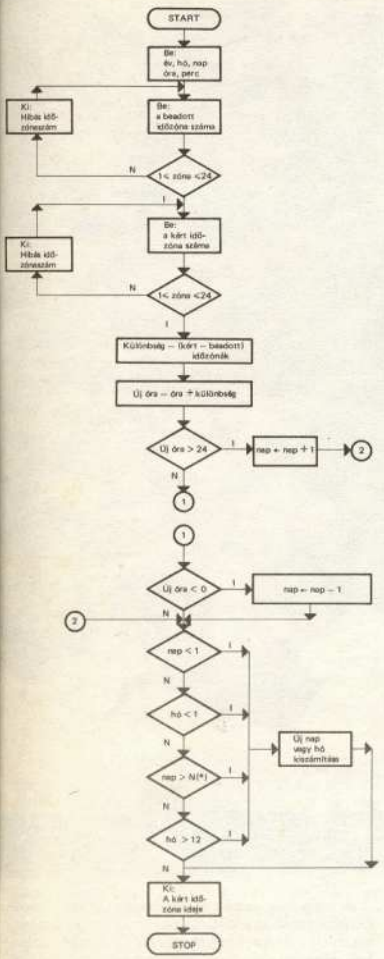
1. Az alábbi program kiírja azokat a természetes számokat, amelyek számjegyeik faktoriálisainak összegével egyenlők:

```
10 A(0)=1:A(1)=1
20 FOR I=2 TO 9
30 A(I)=A(I-1)*I
40 NEXT I
45 REM D-BEN AZ ERTEKTELEN 0-K
   SZAMA VAN
50 D=5
55 PRINT "A SZAMOK:"
60 FOR I=0 TO 9
70 FOR J=0 TO 9
80 FOR K=0 TO 9
90 FOR L=0 TO 9
100 FOR M=0 TO 9
110 FOR N=0 TO 9
120 B=A(I)+A(J)+A(K)+A(L)+
    A(M)+A(N)-D
130 C=N+10*M+100*L+1000*
    K+10000*J*100000*I
140 IF B=C THEN PRINT B
150 NEXT N
160 IF D>4 THEN D=4
170 NEXT M
180 IF D>3 THEN D=3
190 NEXT L
200 IF D>2 THEN D=2
210 NEXT K
220 IF D>1 THEN D=1
230 NEXT J
240 D=0
250 NEXT I
260 PRINT "VEGE"
270 END

A SZAMOK: 1, 2, 145, 40585
```

2. Az időzóna-átszámító program: és a program folyamatábrája a következő:

```
10 REM IDOZONA PROGRAM
12 REM **********************
15 REM * T A P   3 4 - E N *
18 REM **********************
20 DIM A(12): FOR I=1 TO 12
30 READ A(I): NEXT I
35 PRINT "KEREM A DATUMOT"
40 INPUT "EV";E: S=E-INT (E/4)*4
45 IF S<>0 THEN A(2)=28
50 INPUT "HO";H
55 IF H<1 OR H>12 GOTO 50
60 INPUT "NAP";N
65 IF N<1 OR N>A(H) GOTO 60
70 INPUT "ORA";O
75 IF O<0 OR O>23 GOTO 70
80 INPUT "PERC";P
85 IF P<0 OR P>59 GOTO 80
90 INPUT "A DATUM IDOZONAJA";I
95 IF I<1 OR I>24 GOTO 90
100 PRINT "MELYIK IDOZONABAN"
105 INPUT "KERI A DATUMOT";J
110 IF J<1 OR J>24 GOTO 100
120 K=J-I: U=0+K
125 IF U<0 OR U>23 GOTO 150
130 PRINT "A KERT DATUM:"
135 PRINT "EV HO NAP ORA PERC"
140 PRINT E;H;N;U;P
145 PRINT "* VEGE *": END
150 IF U>24 GOTO 170
155 N=N-1: U=U+24: IF N<1 GOTO 180
160 GOTO 130: REM NYOMTATAS
170 N=N+1: U=U-24: IF N>A(H) GOTO 200
175 GOTO 130: REM NYOMTATAS
180 IF H<>1 GOTO 190
185 H=12: N=31: E=E-1: GOTO 130
190 N=A(H-1): H=H-1: GOTO 130
200 IF H<>12 GOTO 220
210 H=1: N=1: E=E+1: GOTO 130
220 H=H+1: N=1: GOTO 130
500 DATA 31,29,31,30,31,30
510 DATA 31,31,30,31,30,31
```



3/a. A programot az alábbi feladatok megoldásához tudjuk felhasználni:

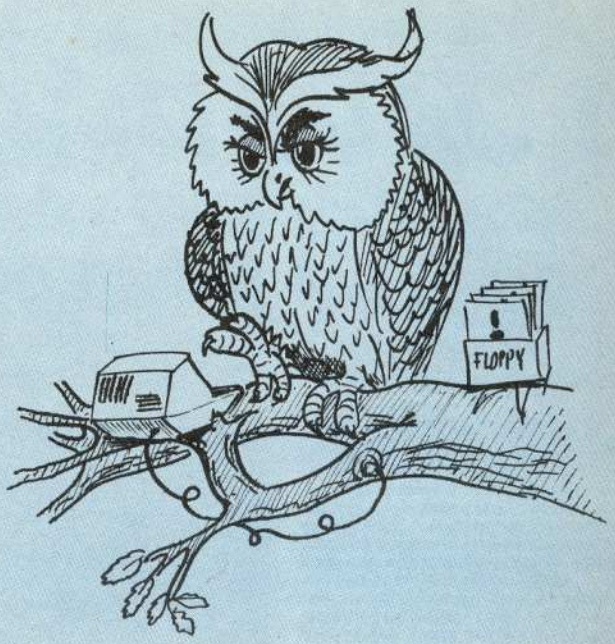
- egy számról el kell dönteni, hogy prímszám-e;
- két szám között van-e prímszám;
- egy intervallumban melyik a legkisebb, illetve melyik a legnagyobb prímszám.

```
5 REM PRIMKIIRATO PROGRAM *SZITA*
10 DEFDBL A,B,C
20 INPUT A:CLS
30 B=INT(SQR(A)): A=A+2
40 IF(A/3)-INT(A/3)=0 THEN 30
50 FOR I=5 TO B STEP 6
60 C=A/I
70 IF C-INT(C)=0 THEN 30
80 C=A/(I+2)
90 IF C-INT(C)=0 THEN 30
100 NEXT I:PRINT A;: GOTO 30
```

3/b. A számítógép által kiírt kérdőjelre meg kell adnunk azt az értéket, amelytől kezdve kérjük a prímszámokat (a szám nem lehet páros!). Az első prímeket nem írja ki (2, 3).

3/c. Az 50-es sorban a lépésközt változtathatjuk meg. Az eredeti programban a lépésköz 2 volt.

A bölcsesség nem boszorkányság



Rövid és ravasz programok

Próbáljuk ki! Nemcsak rajzol, zenél is!

```
100 REM ************************
110 REM *                      *
120 REM *  ABSZTRAKT MUVESZET  *
130 REM *  DINNYES JANOS 1983  *
140 REM *  ERDELYI TIBOR 1984  *
145 REM *     COMMODORE 64     *
150 REM *                      *
160 REM ************************
170 REM
180 POKE53281,8:POKE53280,8
190 PRINT"■"
200 PRINT"■"
210 FOR I=1 TO 24
220 FOR J=1 TO 40
230 PRINTCHR$(205.5+RND(1));
240 NEXT
250 NEXT
260 LQ=LQ+1
270 Q=INT(RND(1)*10):W=
    INT(RND(1)*16)
280 IFLQ<6THEN310
290 LQ=0
300 POKE53281,Q:POKE53280,W
310 I=INT(29*RND(1))
320 J=INT(29*RND(1))
330 K=INT(13*RND(1))
340 L=INT(13*RND(1))
350 M=INT(10*RND(1))
360 GOSUB 510
370 IFM>5THENGOSUB410
380 IFM<6THENGOSUB460
390 Z=INT(15*RND(1))
400 GOTO260
410 FORX=ITOJ
420 FORY=KTOL
430 POKE1024+X+40*Y,32:
    POKE55296+X+40*Y,2
440 NEXTY,X
450 RETURN
460 FORX=ITOJ
470 FORY=KTOL
480 POKE1024+X+40*Y,160:
    POKE55296+X+40*Y,Z
490 NEXTY,X
500 RETURN
510 WW=INT(3*RND(1)+3)
520 FOR IH=1 TO WW
530 WQ=INT(140*RND(1)+10)
540 WE=INT(100*RND(1)+50)
550 S=54272
560 POKE S+1,WQ
570 POKE S+5,9
580 POKE S+6,9
590 POKE S+24,15
600 POKE S+4,17
610 FOR D=1 TO WE: NEXT D
620 POKE S+4,16
630 NEXT IH
640 RETURN
```

JÁTÉK VOLT A JAVÁBÓL!

Stílszerű köszöntés

Március 16-án Szekszárdon lezajlott a lapunk 1983-as számában meghirdetett játékprogram-pályázat döntője. A nyertesek és helyezettek névsorát már közöltük; most magáról az eseményről számolunk be.

Közhírré tettük

Amikor egy pályázaton közeledik a pályaművek beküldési határideje, leginkább a meghirdető izgalma fokozódik: megnyerte-e a „bajnokság" ötlete a potenciális versenyzők tetszését, lesz-e elég résztvevője a versenynek?

E drukknak mi, Magazinunk első pályázatának meghirdetői sem voltunk híján, és – úgy éreztük – nekünk különösen megvolt erre minden okunk. A játékprogram-pályázatot hírül adó kis cikkünk – utólag már mi is tudjuk – kicsit elbújt az egyik lap alján, és meg sem nevezte a díjakat. Még nagyobb baj volt azonban, hogy sok olvasóhoz lapunk csak kevéssel a határidő lejárta előtt, vagy uram bocsá', az után jutott el. Aztán volt egy elég szigorú megkötésünk, hogy a beérkező pályaművek közlésére a Magazin jogot szerez.

Ezek után január 15-én a megkönnyebbülés és öröm sóhaja szállt fel a szervezőkből. Különösen, mikor néhányan még telefonon kértek (és kaptak) pár nap haladékot, mivel későn értesültek a pályázatról.

Lajstromba vettük

Február első napjaiban ült össze a zsüri először, és akkor vette kézbe a harmincnyolc középiskolástól beérkezett negyvenöt program leírását és kazettáját. A jeligés borítékok még szigorúan őrizték a pályázók inkognitóját.

A kikötés mindössze ennyi volt: „Új, önálló, BASIC nyelven HT–1080Z, ABC–80 vagy ZX81 gépekre írt játékokat várunk". Nos, mi tagadás, az első menetben több mű kipottyant. A témában jártas zsüritagok ugyanis könnyen felismerték a különféle kiadványokban már megjelentekhez hasonló, sőt egy-két esetben azokkal teljesen megegyező játékokat.

A zsürin belül kisebb csoportok alakultak, amelyek egyrészt géptípusok, másrészt a játék jellege szerint nézték a pályaműveket. A programok egy része Szekszárdra utazott, másik csoportját berzsenyis diákok „vallatták", megint másik kötegével a Magazin „Játékprogramok" rovatának vezetője foglalkozott.

Kiválasztottuk

A következő találkozóra már minden zsüritag, illetve csoport kiválasztotta a maga favoritjait. Összesen tizennégy programot javasoltak döntőbe jutásra. Ezeket együtt végignézték. Mit, végignézték! A komoly zsüritagok önfeledt játékosokká váltak, elmélyülten nyomkodták a figurákat a képernyőn mozgató billentyűket, bosszankodtak az elhibázott találatok miatt, izgatottan számolták elért pontjaikat. Aztán a játékok nyújtotta szórakozás minősítésén túl a mérleg serpenyőjébe tették az egyes programok szakmai értékét is, értékelték a nehézség fokát, a megoldás szellemességét, megtanulhatóságát, látványosságát, hanghatását stb.

Ez aztán a játék!

Parázs viták után kiválasztották a legjobb tíz játékot. Ekkor kerültek elő a lepecsételt jeligés borítékok, és kiderült, hogy a tíz legjobb versenyző személyében az ország szinte minden tája képviselve van.

Hadd kotyogjunk ki egy kedves epizódot. Úgy látszik, a pályázat résztvevői is drukkoltak. „Sárgarigó" borítékjában ott volt a pontos cím (emelettel és ajtóval), de a pályázó neve helyett megint csak ez a szó állt: Sárgarigó. Lehet, hogy a postás ezután így fogja őt nevezni; mindenesetre megtalálta, mert a döntőre megérkezett a versenyző, és sikeresen szerepelt.

A helyszín és a házigazdák

A szekszárdi Mártírok terének tágas négyszögét egyik oldalról tekintélyes, frissen vakolt, vöröstéglás épület zárja le. Első pillanatban kitalálja a városba érkező idegen, hogy iskola; akkor is, ha udvarán, kapuja előtt és lépcsőfeljáróján nem is lát éppen diákot.

A szép hagyományokat ápoló Garay János Gimnáziumban eltöltött rövid idő is az ember középiskolás korából megőrzött legszebb emlékeit ébreszti fel. Nem elsősorban az évenként sorra kerülő, az idén is ragyogóan sikerült Garay-napok rendezvényeire gondolok, ennek látványos és pergő, de ugyanakkor a diákok komoly felkészültségét is tanúsító eseménysorozatára, hanem arra a légkörre, hangulatra, ami az odacsöppent idegent megkapja.

Hozzá tartozik ehhez a ragyogó tisztaság éppúgy, mint a világos, virágos folyosók és tanári szobák, a fantáziával, ötlettel teli feliratok, plakátok, a harminc évvel ezelőtt készült tablók a falakon. Része ennek a több ezer kötetes, igazi kincsszámba menő sorozatokat, régi lexikonokat, de valamennyi most megjelenő irodalmi folyóiratot is kínáló, olvasóterme könyvtár, a gyerekek üzemeltette büfé is. És ami valahogy minden arcot mosolyba öltöztet, az a meghitt, nyugodt hang nemcsak a tanár és tanár, de a diák és tanár között is.

Ez az iskola, ez a tanári kar és ez a diákság adott otthont féltő gonddal előkészített ünnepségsorozata egyik eseményeként a Neumann Társaság kezdeményezte játékprogram-pályázat döntőjének, és látta vendégül két napig a döntőbe jutott tíz versenyzőt. Közülük csak a

Sárgarigót faggatja a zsüri

Ki nyert ma?

Szavaz a közönség

kecskeméti Tóth Péter maradt távol, aki aznap az országos középiskolás tanulmányi verseny döntőjén szerepelt. Programját itt egy garaysta képviselte.

A verseny

Már reggel nyolctól folyamatosan érkeztek nemcsak a résztvevők, hanem a nézők is a kilenc órára meghirdetett eseményre. A szekszárdi középiskolákból és Budapestről, az ELTE Numerikus Gépi Matematikai Tanszékéről, a Kandó Kálmán Főiskoláról, az Országos Pedagógiai Intézettől kölcsön kapott húsz gép három teremben (külön a HT–1080Z-k, az ABC–80-ak és a ZX81-ek), rajtszámokkal ellátva, a közönséget stílszerűen üdvözölve várták a versenyzőket és szurkolókat.

A zsüri elnöke, Kovács Győző, a Neumann János Számítógéptudományi Társaság főtitkára, bemutatta a zsürit (Zentai András, a Garay Gimnázium igazgatója; Bayer József, a gimnázium tanára; Kádár András, az NJSZT Tolna megyei titkára; Dávid Gábor, az MTA SZTAKI főmunkatársa; Kiss Domonkos, a Novotrade Rt. Szoftver Stúdiójának vezetője; Tóth Istvánné, az NJSZT ügyvezető titkára; Juhos Nándor, a Berzsenyi Gimnázium tanulója), ismertette a döntő menetét és a győztesekre váró díjakat. Kérte a közönséget, hogy szavazataikkal támogassák a nekik legjobban tetsző játékokat.

A tíz versenyző két-két gépen tölthette be programját, így több néző tudta egyszerre körbeállni a képernyőket.

Nehéz helyzetben a zsüri

Minden zsüritag 1–10 ponttal jutalmazhatta az egyes játékokat. Így a maximális elérhető érték 80 volt. Mielőtt a pontszámokat közölték volna, a zsürielnök ismertette a játékok programozási megoldásait, a zsüritagok pedig felhívták a figyelmet azok egyéb értékeire, egy-egy jó ötletre vagy hiányosságra. Hogy mennyire igaz a közhely: „a zsüri nehéz helyzetben volt", bizonyítják Kádár András szavai: „Én már jó néhányszor átnéztem ezeket a programokat. Érdekes módon minden alkalommal új dolgokat fedezek fel bennük, és ha eddig többször kellett volna döntenem, hogy melyik a legjobb, alighanem minden esetben másikra adtam volna a voksomat." Az előzetes vita olyannyira nem befolyásolta a szavazást, hogy néhány versenyműre adott pontszám 3–9 között ingadozott. A sorrend összeállítása után bontották fel a közönségszavazatokat tartalmazó urnát. A zsüri elégedetten állapította meg, hogy döntése összhangban volt a közönségével.

A délelőtt önfeledten játszó diákok vérbeli drukkerekként végigvárták a versenyzőkkel az eredményhirdetést. A fizikai előadó zsúfolásig tele volt, pedig már ebédidőre járt. A házigazdák pillanatok alatt a hosszú asztalra varázsolták az értékes díjakat és a szekszárdi fazekes népművész, Fusz Gyula külön ez alkalomra készített, gyönyörű kócsögeit. Ezekből minden versenyző kapott emlékül. Az arcokon izgatott várakozás, miközben Kovács Győző meleg, közvetlen szavakkal, jó tanácsok formájában adta tovább a zsüri döntéshozatala közben elhangzott bírálatokat és hiányosságokat, de természetesen kiemelte az erényeket is. És végül a zsüri eredményt hirdetett.

A díjakat Zentai András adta át a boldog nyerteseknek. Nem tudom, feltűnt-e a kedves Olvasónak a sors iróniája? A tíz döntőbe jutott közül egy versenyző nem tudott eljönni, és ő aratott le mindent! A kétszeres győztes Tóth Péter később, Budapesten, a Magyar Televízió székházában vette át az első díjat.

A zsüri, különdíjként, a döntő egyetlen hölgy résztvevőjének virággal kedveskedett, amelyet a legifjabb diák-zsüritag adott át, elismerő puszik kíséretében.

Utójáték

Előbb a tíz legjobb programot, majd ezt követően a pályázatra érkezettek javát is folyamatosan közöljük lapunkban. Nagyon kíváncsiak vagyunk olvasóink véleményére. Ezért abban a számban, amelyik az utolsó programot hozza, közzéteszünk egy szavazócédulát, amelyen mindenki megírhatja, melyik játékot tartja legjobbnak. A legtöbb szavazatot kapott játék szerzője az „Olvasó díja"-ként egy sétálómagnót kap.

A szekszárdi Garay János Gimnázium és Óvónőképző Szakiskola igazgatója, Zentai András a gimnázium diákjainak és tanárainak megbízásából 1984. március 31-én ünnepélyesen felkérte a Neumann János Számítógéptudományi Társaságot és a Mikroszámítógép Magazin szerkesztőségét, hogy évenként rendezze meg a játékprogram-pályázatot, és felajánlotta, hogy hasonlóan az ez évihez, március 16-án, a Garay-napok alkalmából otthont ad a pályázat döntőjének, szívesen látja a pályázókat és a zsüri tagjait.

A szíves felkérésre nem tudtunk nemet mondeni. Így: játékprogram-mesterek, „vigyázó szemetek Szekszárdra vessétek"!

PÁKOZDI

Az első díjas játékprogram:

A SAJT, AZ EGÉR, A MACSKA ÉS A GAZDASSZONY

A játékprogram versenyen – amit lapunk legelső száma hirdetett meg – 10 játékprogramot díjazott a zsüri és a szekszárdi diákokból álló közönség.

Rovatunkban sorra közölni fogjuk a nyertes programok listáját.

Íme az első: szerzője Tóth Péter, a kecskeméti Katona József Gimnázium IV. oszt. tanulója. A gép: ABC–80.

A játékötlet eredeti és szellemes, a grafika kellemes és olyan gyors, hogy még gyakorlott játékosnak is össze kell szednie magát, hogy egy erősebb fokozatban ne kapjon ki a gazdasszonytól.

A játék lényege ugyanis, hogy a játékosnak, mint macskának az ideoda cikázó egeret be kell kapnia. Amíg ez nem sikerül, addig az egér a képernyő közepén lévő jókora sajtot tünteti el. Ha gyenge egérfogó a cica, a gazdasszony jól elveri, és kezdődik elölről a játék.

Gratulálunk Tóth Péternek!

```
10 REM Jelszo :        G.
20 REM GA(zdasszony)M(acska)E(gèr)S(ajt)
30 RANDOMIZE
40 ; CHR$(12);
50 DIM N$(11)=8,M$(3,3)=7,V$(3,6)=7
60 GOSUB 900
70 GOSUB 1700
80 GOSUB 1980
90 FOR J%=3% TO 18% : FOR I%=12% TO 27% : POKE FN
   P(J%,I%),127% : NEXT I% : NEXT J%
100 A=0 : B=38 : C=21 : D=1 : M9=1
110 RANDOMIZE
120 REM itt kezdôdik
130 W8=W : W9=W1
140 IF RND>.5 THEN W=1 ELSE W=2
150 IF RND>.5 THEN W1=1 ELSE W1=2
160 Z=INT(RND*15) : FOR X=0 TO Z : ON W GOSUB 210
   ,290 : F1=1 : GOSUB 500
170 IF INT(RND*15)=3 THEN GOSUB 370
180 NEXT X : GOTO 120
190 IF PEEK(FNP(A,B))=127 THEN OUT 6,0 ELSE OUT 6
   ,241
200 RETURN
210 ; CUR(A,B)" "; : ON W1 GOTO 220,260
220 A=A-1 : IF A=-1 THEN A=0
230 GOSUB 190
240 ; CUR(A,B)"-";
250 RETURN
260 A=A+1 : IF A=24 THEN A=23
270 GOSUB 190
280 ; CUR(A,B)"'"; : RETURN
290 ; CUR(A,B)" "; : ON W1 GOTO 300,340
300 B=B-1 : IF B=0 THEN B=1
310 GOSUB 190
320 ; CUR(A,B)"x";
330 RETURN
340 B=B+1 : IF B=39 THEN B=38
350 GOSUB 190
360 ; CUR(A,B)"t"; : RETURN
370 Y7=SQR((A-C)^2%+(B-D)^2%)
380 IF Y7>3 THEN 420
390 IF W=1 THEN W=2 ELSE W=1
400 IF W1=1 THEN W1=2 ELSE W1=1
410 X=0 : Z=5 : GOTO 490
420 D9=A-11 : D8=B-20
430 IF D9>8 OR D9<-8 THEN 450
440 IF D8<8 AND D8>-8 THEN 490 ELSE 470
450 IF D9>8 THEN W=1 : W1=1 : X=0 : Z=9 : GOTO 49
   0
460 IF D9<-8 THEN W=1 : W1=2 : X=0 : Z=9 : GOTO 4
   90
470 IF D8>8 THEN W=2 : W1=1 : X=0 : Z=9 : GOTO 49
   0
480 IF D8<-8 THEN W=2 : W1=2 : X=0 : Z=9 : GOTO 4
   90
490 RETURN
500 REM a matska ôm-
510 M=INP(56) : IF M>128 THEN 530
520 FOR L9=1 TO 20 : NEXT L9 : ; CUR(C,D)M$(M9,1)
   CUR(C+1,D)M$(M9,2); : RETURN
530 IF M=199 OR M=232 THEN 590
540 IF M=200 OR M=233 THEN 640
550 IF M=217 OR M=249 THEN 690
560 IF M=194 OR M=226 THEN 750
570 IF F7<F8 THEN GOSUB 830
580 GOTO 520
590 REM balra
600 M9=2 : D=D-1 : IF D=0 THEN D=1
610 IF (PEEK(FNP(C,D)) OR PEEK(FNP(C+1,D)))=127 T
   HEN F1=0 : D=D+1
620 ; CUR(C,D);M$(2,1)CUR(C+1,D)M$(2,2); : IF F1=
   1 THEN ; CUR(C,D+6)" "CUR(C+1,D+6)" ";
630 F7=0 : RETURN
640 REM Jobbra
650 M9=1 : D=D+1 : IF D=34 THEN D=33
660 IF (PEEK(FNP(C,D+5)) OR PEEK(FNP(C+1,D+5)))=1
   27 THEN F1=0 : D=D-1
670 ; CUR(C,D)M$(1,1)CUR(C+1,D)M$(1,2); : IF F1=1
    THEN ; CUR(C,D-1)" "CUR(C+1,D-1)" ";
680 F7=0 : RETURN
690 REM fel
700 C=C-1 : IF C=-1 THEN C=0
710 T1=FNP(C,D) : FOR T=T1 TO T1+5 : IF PEEK(T)<>
   127 THEN NEXT T
720 IF T<T1+6 THEN F1=0 : C=C+1
730 ; CUR(C,D);M$(M9,1)CUR(C+1,D)M$(M9,2); : IF F
   1=1 THEN ; CUR(C+2,D)"      ";
740 F7=0 : RETURN
750 REM le
760 C=C+1 : IF C=22 THEN C=21
770 T1=FNP(C+1,D) : FOR T=T1 TO T1+5 : IF PEEK(T)
   <>127 THEN NEXT T
780 IF T<T1+6 THEN F1=0 : C=C-1
790 ; CUR(C,D);M$(M9,1)CUR(C+1,D)M$(M9,2); : IF F
   1=1 THEN ; CUR(C-1,D)"      ";
800 F7=0 : RETURN
810 IF (C>1 AND C<19) AND (D>6 AND D<28) THEN F1=
   1 ELSE F1=0
820 RETURN
830 REM vizsgálat
840 F7=F7+1
850 E=A-C : F=B-D : ON M9 GOTO 880,860
860 ; CHR$(7)CUR(C+1,D)".."; : IF (E=0 OR E=1) AND
    (F=0 OR F=1) THEN 1000
870 GOTO 890
880 ; CHR$(7)CUR(C+1,D+5)"-"; : IF (E=0 OR E=1) A
   ND (F=4 OR F=5) THEN 1000
890 RETURN
900 DATA "fpppzt","  7777$","xuppp9","$kkkk "
910 DATA "  èôà 5","  z■■ 5","  7z5 u","  h■=&7",
   " *■5 5","  h■u !"
920 DATA "  ô0 ","  h■■■u ","  ô■■■■ ","  /o/o/ ",
   "   JpJp "
930 DATA " 5     "," 5     ","èà     ","&7     "," 5
     "," !    "
940 DATA "       ","     p ","  è>!  ","1ô'    ",".!
     ","      "
950 DATA "       ","       ","       ","1      ","#,
   p    ","   $,p"
960 READ M$(1,1),M$(1,2),M$(2,1),M$(2,2) : FOR J=
   0 TO 10 : READ N$(J) : NEXT J
970 FOR G=1 TO 3 : FOR G1=1 TO 6 : READ V$(G,G1)
   : NEXT G1 : NEXT G
980 DEFFNP(A%,B%)=31744%+A%/8%*40%+(A%-A%/8%*8%)*
   128%+B%
990 RETURN
1000 REM itt következik a vègkifejlet
```

```
1010 N=0 : P=0 : OUT 6,0
1020 FOR J%=3% TO 18% : FOR I%=12% TO 27% : P=P+1
1030 IF PEEK(FNP(J%,I%))=127% THEN N=N+1 ELSE POK
    E FNP(J%,I%),255
1040 NEXT I% : NEXT J%
1050 FOR T=0 TO 5000 : NEXT T
1060 ; CHR$(12);N/P*100"%-a maradt meg a sajtnak.
    "; : OUT 6,135
1070 FOR K=1 TO 3500 : NEXT K
1080 IF N/P>.75 THEN 1640
1090 IF N/P>.5 THEN 1490
1100 IF N/P>.25 THEN 1350
1110 REM macska puff
1120 ; : ; : ; "    Ez nagyon gyenge eredmény.Gon
    dolj    bele! nem kellene felhagyni a "
1130 ; " macska-    sággal?" : ; : ; " De ugy láts
    zik, nemcsak én látom igy!";
1140 FOR L=1 TO 8000 : NEXT L
1150 GOSUB 1980 : ; CUR(7,30)M$(2,1)CUR(8,30)M$(2
    ,2);
1160 FOR J=0 TO 10 : ; CUR(J+5,1)N$(J);
1170 ; CUR(J+5,7)" "; : NEXT J
1180 FOR J=11 TO 18 : SETDOT 23,J : NEXT J
1190 FOR I=0 TO 30 : FOR J=11 TO 18 : CLRDOT 23,J
    : NEXT J : FOR J=15 TO 29 : SETDOT J,14 : NEX
    T J
1200 OUT 6,133 : FOR K9=0 TO 50 : NEXT K9
1210 FOR J=15 TO 29 : CLRDOT J,14 : NEXT J : FOR
    J=11 TO 18 : SETDOT 23,J : NEXT J
1220 OUT 6,0 : FOR K9=0 TO 50 : NEXT K9 : NEXT I
1230 FOR L=1 TO 30 : FOR K=1 TO 50 : NEXT K : OUT
    6,137
1240 09=INT(RND*5)+24 : FOR A9=09 TO 09+30 STEP 5
    : SETDOT 23,A9 : CLRDOT 23,A9-5
1250 NEXT A9 : CLRDOT 23,A9-5
1260 K8=INT(RND*6)+21 : K9=INT(RND*12)+60
1270 09=INT(RND*5)+40 : SETDOT 23,09 : CLRDOT 23,
    09
1280 IF DOT(K8,K9) THEN CLRDOT K8,K9 ELSE SETDOT
    K8,K9
1290 OUT 6,0 : NEXT L
1300 FOR K=60 TO 71 : FOR L=26 TO 21 STEP -1 : IF
    DOT(L,K) THEN GOSUB 1320
1310 NEXT L : NEXT K : FOR K9=1 TO 1000 : NEXT K9
    : GOTO 1340
1320 FOR L5=L+1 TO 47 : IF DOT(L5,K)=0 THEN SETDO
    T L5,K : CLRDOT L5-1,K : NEXT L5
1330 RETURN
1340 ; : ; : ; : ; : END
1350 REM macska elk.
1360 ; : ; : ; : ; : ; " Ilyen macska nem is kell
    !" : ; : ; " Menj, amerre látsz, de gyorsan!"
1370 FOR I=0 TO 3000 : NEXT I : ; : ; : ; " Mès m
    indis itt vagy?" : FOR I=0 TO 2000 : NEXT I
1380 GOSUB 1980 : FOR K=1 TO 24 : FOR K1=0 TO 10
1390 ; CUR(K1+5,K)N$(K1); : NEXT K1 : ; CUR(14,K+
    7)" "M$(1,1);CUR(15,K+8)M$(1,2); : NEXT K : GO
    SUB 1980
1400 FOR L=25 TO 5 STEP -1 : FOR J=1 TO 5 : ; CUR
    (J+4,L)"■■ "; : NEXT J : ; CUR(9,L-3)" ";
1410 FOR J=0 TO 25 : NEXT J : NEXT L
1420 GOSUB 1980 : FOR L=10 TO 20 : ; CUR(5,L)" "
    CUR(6,L)" "CUR(7,L)" èÞ";
1430 FOR J=0 TO 50 : NEXT J : NEXT L
1440 GOSUB 1980 : FOR L=18 TO 13 STEP -1 : ; CUR(
    5,L)"Þ  ";
1450 FOR J=0 TO 100 : NEXT J : NEXT L
1460 GOSUB 1980 : FOR L=14 TO 16 : ; CUR(5,L)" 50
    "; : FOR J=0 TO 150 : NEXT J : NEXT L
1470 GOSUB 1980
1480 END
1490 REM macska elv.
1500 ; : ; : ; : ; "  Visszàzhattàl volna jobban i
    s arra a    sajtra!"
1510 ; : ; " NA MAJD MASKOR MEGTANULOD!!!"
1520 FOR T=0 TO 5000 : NEXT T
1530 ; CHR$(12); : FOR G=0 TO 22 : ; CHR$(151) :
    NEXT G : ; CHR$(151);
1540 FOR K=1 TO 24 : FOR K1=0 TO 10
1550 ; CUR(K1+5,K)N$(K1); : NEXT K1 : ; CUR(14,K+
    7)" "M$(1,1);CUR(15,K+8)M$(1,2); : NEXT K
1560 FOR V=1 TO 25 : FOR G1=1 TO 3 : FOR G=1 TO 6
    : ; CUR(G+4,29)V$(G1,G); : NEXT G
1570 FOR T=0 TO (4-G1)*20 : NEXT T : NEXT G1 : OU
    T 6,77 : FOR T4=0 TO 9 : NEXT T4 : ; CHR$(7);
1580 ; CUR(14,32)"■9///%+"CUR(15,32)"hhhhô";
1590 FOR T=1 TO 150 : NEXT T : FOR G=1 TO 6 : ; C
    UR(G+4,29)V$(2,G); : NEXT G
1600 FOR T=0 TO 50 : NEXT T : ; CUR(14,32)M$(1,1)
    CUR(15,32)M$(1,2);
1610 ; CUR(4,28)CHR$(135)V;CHR$(151); : NEXT V :
    ; CHR$(7);CHR$(7);
1620 FOR K=14 TO 15 : FOR L=32 TO 37 : POKE FNP(K
    ,L),PEEK(FNP(K,L))+128 : NEXT L : NEXT K
1630 ; : ; : ; : ; : ; : ; : ; : ; : ; : ; : ; :
    ; : ; : END
1640 REM macska marad
1650 ; : ; : ; : ; " Ügyes cica vagy ."
1660 ; " Biztos véletlen volt."
1670 ; : ; : ; " Most az egyszer mès maradhatsz"
    : FOR K=1 TO 8000 : NEXT K : GOTO 80
1680 OUT 6,1
1690 END
1700 REM tàjèkoztato
1710 FOR J=1 TO 50 : ; CHR$(151) : NEXT J
1720 POKE FNP(1,2),230,230,230,230,230,230,230,23
    0,230,230,230,32,84,93,74,64,75,79,90,84,65,84
    ,79
1730 POKE FNP(1,25),32,230,230,230,230,230,230,23
    0,230,230,230
1740 ; CUR(23,0)CHR$(151);
1750 ; CUR(3,2)" A JATÉKBAN A MACSKANAK EL KELL"C
    UR(4,3)"FOGNIA AZ EGERETè";
1760 ; CUR(6,3)"AZ EGER A SAJT KÖRÜLB VAGY A SAJT
    BA"CUR(7,3)"RAGOTT UTAKON SZALADGAL";
1770 ; CUR(9,3)"IGYEKEZZ MINEL TÖBB SAJTOT"CUR(10
    ,3)"MEGÖRIZNIè";
1780 ; CUR(12,3)"A MACSKA CSAK A FEJE FELÖL TUDJA
    ";
1790 ; CUR(13,4)"ELKAPNI AZ EGERETè";
1800 ; CUR(15,3)"A MACSKA A KÖVETKEZÖ GOMBÖKKAL"C
    UR(16,3)"IRANYITHATOb "CUR(16,24)"S A TÖBBI";
1810 POKE FNP(16,16),217,96,194,96,199,96,200
1820 ; CUR(17,3)"GOMBBAL PEDIG EL LEHET"CUR(18,3)
    "KAPNI AZ EGERETè";
1830 ; CUR(20,13)"?//////o"CUR(22,13)"àööööööö"CU
    R(21,4)"A JATÉK A5RETURNJGOMBBAL INDUL";
1840 ; CUR(9,30)M$(2,1)CUR(10,30)M$(2,2);
1850 FOR K=20 TO 22 : FOR L=13 TO 20 : POKE FNP(K
    ,L),PEEK(FNP(K,L))+128 : NEXT L : NEXT K
1860 POKE 31745,230
1870 A=INT(RND*37)+1 : B=INT(RND*20)
1880 IF PEEK(FNP(A,B))<>32 THEN 1870
1890 POKE 31745,32
1900 Z=INT(RND*20) : W=INT(RND*2)-1 : W1=INT(RND*
    2)-1
1910 IF W THEN B1=0 : IF W1 THEN A1=-1 ELSE A1=1
    ELSE A1=0 : IF W1 THEN B1=-1 ELSE B1=1
1920 W1=W1+2 : W=W+2
1930 FOR A=0 TO Z : IF PEEK(FNP(A+A1,B+B1))<>32 T
    HEN 1900 ELSE ON W GOSUB 210,290
1940 FOR K=0 TO 20 : NEXT K
1950 IF INP(56)=141 THEN 1990 ELSE IF INP(56)>128
    THEN 1970
1960 NEXT A : GOTO 1900
1970 ; CUR(10,30)CHR$(7)".";  : FOR K=1 TO 50 : NE
    XT K : ; CUR(10,30)"Þ"; : GOTO 1960
1980 ; CHR$(12); : FOR I=0 TO 23 : ; CUR(I,0)CHR$
    (151); : NEXT I : RETURN
1990 ; CHR$(12)"MELYIK FOKOZATTAL INDULJUNK?( 1-9
    )";
2000 ; : ; : ; "( Az egyes számok azt jelentik, h
    ogy     egy  adott helyről - elmozdulás
      nélkül - ";
2010 ; "hányszor próbálhatja el-       kapni a ma
    cska az egeret          )";
2020 ; "      / az 1-es a legnehezebb /      ";
2030 GET T$ : S7=ASC(T$)
2040 IF S7<49 OR S7>57 THEN 1990
2050 F8=VAL(T$) : RETURN
```

Éljen a kedves Olvasó!

De bocsásson is meg, hogy a levelekre meglehetősen lassan válaszolok. Nagyon sokat kaptam, több, mint kétszázat! Meg kell szerveznünk a válaszadást, kollektivizálnunk kell a levelezést. Addig is olvassák és élvezzék a bíráló, javaslatot tevő leveleket. Érdemes.

Baukó János, Békéscsaba,

Majakovszkij u. 31. 5600

Már több levelet írtam a Társaságnak: hármat Önnek és egyet Simonyi Endrének. A levelek tárgya: számítógép-kapcsolási rajz, nyomtatási rajz, alkatrészlista, műszaki vélemény, leírás.

Sajnos egyik levélre sem kaptam választ. Tehát ezúton kérem Önt és Simonyi Endrét, küldjenek számunkra (Gera, Sári, Bankó) valami kapcsolási rajzot. Vagy a Kalendárium című tv-adásban (még ősszel) bemutatott Házi Computer Club kisszámítógépét, amelyet igen egyszerűnek és ötletesnek találtunk, és amely szintén a Lukács testvérek találmánya, mint az AIRCOMP–16, melynek szintén szívesen vennénk a kapcsolási rajzát. A gépet azért akarjuk megépíteni, hogy tanulhassunk. Joggal kérdezhetik, hogy az iskolában miért nem elég a számítógépes foglalkozás. Hát nem elég! Meg fog döbbenni: iskolánkban kéthetente van számítógépes szakkör, és minden foglalkozás pontosan 1 órás!!! Így beláthatják, miért adjuk számítógép-építésre a fejünket. Mellesleg én a nyáron elvégeztem egy programozási előkészítőt, és szeretném folytatni, amit elkezdtem, de ilyen körülmények között nem lehet.

A géppel nem az a célunk, hogy felvágjunk, hanem tanulni akarunk. Ezért kérjük az Önök segítségét, amit előre is köszönünk. Ha mégsem tudnak eleget tenni a kérésünknek, kérem, írja meg, hol férhetünk hozzá ilyesmihez, vagy mikor lesz a µM-ban ilyen tárgyú cikk.

Ha mégsem kapunk választ, nem hagyjuk annyiban. Addig írunk, míg ki nem fogy a tinta a tollbetétünkből. És ha mégis kifogy, hát újat veszünk és folytatjuk a zaklatást. A levelet nagyon várjuk.

Ui. A levélben mellékelünk egy fényképet attól az alkotótól, aki a levélben szerepel, és arról a számítógépről, amit meg szeretnénk építeni. A fénykép az IM 1983. októberi számában jelent meg.

Haragos szavai termő talajra hulltak: a sorozatot ebben a számban megkezdtük! Amíg nem tudtunk mit válaszolni, addig formális választ nem akartunk küldeni.

Warga Éva és Szabó Attila, Damjanich J. Gimn.

Kunszentmiklós, 6090

Szeretnénk megkérdezni, hogy a HT–1080Z iskolaszámítógéppel lehet-e órarendkészítő programot csinálni. (Az iskolánkban mindössze 8 osztály van, minden évfolyamból kettő.) Ha lehet ilyen programot írni a gépre, kérjük, írják meg, hogyan fogjunk hozzá. A segítséget és az információkat előre is köszönjük. Első gimnazisták vagyunk, most ismerkedünk a számítógéppel, és a jelenlegi órarend nem nagyon tetszik nekünk.

Válaszoljon, akinek van ilyen programja! Küldje el nekünk is, ha jó, közöljük. Hivatalosan ilyen program nem kapható.

Tevan Imre, Budapest,

Lóránt u. 8/a. 1125

Nagyon szívesen írnék én is cikkeket a µ M-ba, ha témáimat megfelelőnek tartják:

1. MO8X kontra Commodore–64. Az ötletet a cikkhez saját rossz tapasztalataim adták. Meggyőződésem ugyanis, hogy sok intézmény rosszul választ személyi számítógépet, „hasra esve" a nyugati játékgépek egyébként csodálatos, színes grafikája, hanggenerátora stb. előtt, és tájékozatlanság, butaság, sznobság, hozzá nem értés miatt megfeledkezik az MO8X azon előnyös tulajdonságairól, amelyek komoly, tudományos munkák elvégzésére sokkal alkalmasabbá teszik például a VC–64-nél.

2. BASIC az MO8X-en (eltérések, bővítések, trükkök).

Gondolkoztam, hogy illik-e közölnöm azt a levelet, amelyben – az SZKI munkatársa lévén – érdekelt vagyok. Mivel a dicséret munkatársaimnak és a gép konstruktőreinek szól, ezért úgy éreztem, álszerénység lenne elhallgatnom. Köszönjük.

Jenei Zoltán, Békés,

Meggy u. 16.

A Stúdió '84-ben sugárzott riport kapcsán szeretnék felvilágosítást kérni. Villámcsapásként hatott rám Önnek az a mondata, hogy a társaság – nem tudom, hogy a HCC vagy az NJSZT – számítógép-építést tervez, amelynek keretében biztosítaná az alkatrészeket, amelyeket otthon csak össze kellene forrasztani.

Itt Békésen az a hír járja – talán rémhírnek is lehetne nevezni –, hogy egy ASCII billentyűzet 3000 forintba kerül. Szerintem ez csak a hall-generátoros típusokra igaz, de váltig állítják, hogy ez tisztán mechanikus kivitelű. Ha pedig ez igaz, akkor gondolkodóba esik az ember, hogy otthon, házilag készítsen-e ilyet, vagy megvegye. Ön biztosan többet tud erről, és jó volna eldönteni ezt a vitát!

1. A µ M-ban az egyik gép ismertetését megkezdtük, ehhez az alkatrészt mindenkinek magának kell beszerezni. A másikat most készítjük elő, remélem sikerül az alkatrészeket, NYÁK-ot, klaviatúrát biztosítani.

2. A hír – sajnos – igaz.

Tóth Ákos, Siklós,

Sallai u. 6. 7800

A FORTH nyelvről írtak alapján úgy gondolom, hogy igen ügyes nyelv lehet. Erről valami részletesebb, konkrét leírás lesz-e, vagy hol lehet hozzáférni? Esetleg egy Z80-as alapú gépre adaptálható FORTH operációs rendszer nincs-e birtokukban? Ugyanis szívesen foglalkoznék ezzel a leírásuk alapján igen rugalmasnak, jól használhatónak tűnő nyelvvel! A Z80 utasításkészlete (magyar nyelvű leírással együtt) birtokomban van. Ha netán szükségük lenne rá, kérem, jelezzék!

Érdekelnének az NJSZT Hobbi Szakosztálya által készített kártyák, a ROM és RAM bővítés, valamint hogy ezeken kívül mi van még? És természetesen a kártyák árát is szeretném megtudni, mivel rendelnék belőlük. Sajnos Budapesttől igen messze lakom, de szeretnék jelentkezni a Társaságba és a HCC-ba is. A Társaság és a Klub mit tud nyújtani a vidéki „pártoló" tagoknak? A rendezvényeken csak igen ritkán tudnék részt venni, de levélen keresztül tarthatnánk a kapcsolatot. Kérem, erre is válaszoljanak!

1. A FORTH nyelvről Pataki Ernő ír könyvet, a megjelenéséről tudósítunk.

2. A Z80 utasításkészletet küldje el, megnézzük, hátha közölni tudjuk.

3. A HCC szívesen terjeszkedne. Hozzanak létre helyi HCC-t, mi készséggel támogatjuk a tagokat, segítjük az alkatrészek beszerzését. A kártyákra a HCC válaszol.

Wessely István gépészmérnök, Vecsés,

Somogyi Bacsó u. 36. 2220

Örömmel olvastam a µ M első számát. Több mint 14 éves programozási gyakorlatom van, de mégis találtam benne újat, érdekeset. Különösen a FORTH rendszer ismertetése tetszett.

A számítógép történelme nemcsak Sajni Józsefet érdekli, hanem engem is. Tavaly szükségem lett volna a témában egy jó tanulmányra vagy cikkre, de nem találtam. Több szakkönyv és tankönyv bevezető fejezetében található néhány adat és rövid összefoglalás, de nem volt időm ezeket felkutatni és feldolgozni. Ha nincs vállalkozó a cikk megírására, szívesen foglalkoznék vele.

Igen fontos a hobbigép-készítés támogatása, de én nem vagyok elektromos érdeklődésű, és valószínűleg sohasem fogok saját hobbigépet készíteni. Az alkalmazás az érdeklődési körön. Ezért Gulyás László olvasóval ellentétben én nem az alkatrészellátás biztosítására várok, hanem arra a gépre, amit néhány heti keresetemből meg tudok venni. Remélem, nem tart túl sokáig.

Verhás Péter figyelmét szívesen felhívnám arra, hogy a Lőcs–Sarkadi–Nagy–Szlankó-féle BASIC könyv keresése helyett válogasson a most megjelent BASIC könyvekből. Az említett kiadvány évekkel ezelőtt jelent meg, jó szakkönyv, de stílusa nem a mikroszámítógépeseké, hiszen akkor még nem is nekik íródott.

Az 1983. évi számban a FORTH cikk „tetszési indexe" volt a legmagasabb. Pataki Ernő nevében is köszönjük az elismerést. A számítástechnika történetéről a cikksorozatot elkezdtük. Reméljük, az olcsó gép is hamarosan megvalósul; igyekszünk szerény lehetőségeinkkel segíteni.

Csumpilla Tivadar, Pilisborosjenő,

Sallai I. u. 9. 2097

Levelét a Híradástechnika Szövetkezethez továbbítottuk.

Péter József honvéd, Szentendre,

pf. 166/A, 2001

Egerben, az Alpári Gyula Közgazdasági Szakközépiskolában végeztem, számítástechnikai–programozó tagozaton. Jelenleg sorkatonai szolgálatomat töltöm. A középiskolában több alkalommal részt vettem különböző számítástechnikai vetélkedőkön, versenyeken,

amelyeket az Önök Társasága rendezett. Amennyiben van rá lehetőség, hogy sorkatonaként (mondjuk ilyen kategóriában) is „versenyezhessünk", kérem, értesítsen ideiglenes címemen.

Örömmel közölhetem, hogy megalakult a Honvédelmi Szakosztály, amelynek egyik célja a számítástechnika lehetőségeinek megteremtése a helyőrségekben, elsősorban a sorkatonák részére. A helyi μ klubok megalakítása időt vesz igénybe, de jó tudni, hogy vannak, akikre már számíthatunk.

Masa István, Budapest,

Bartók Béla út 17. 1114

TI 99/4A típusú személyi számítógépem van, és ehhez most egy SEIKOSHA GP–100 VC típusú nyomtatót kaptam, amit nem lehet a számítógéphez közvetlenül csatlakoztatni. Megoldás lenne számomra az is, ha a számítógépem például egy VIC–20-ra cserélném. Ehhez kérem az Önök segítségét.

Lehet jelentkezni!

Veress József gimn. tanár, Mátészalka,

Fürst S. u. 57. 4700

Azt kéri, hogy jegyezzük elő a „zacskós számítógép"-re, ha az akció megindul.

Dolgozunk, hogy legyen, és örök optimistaként nagyon remélem, hogy a terv sikerül, és sikerül egy 6–8000 forintos, 16 k-s kitet összehoznunk. Azt szeretnénk, ha az első darabok május–júniusban, a nagyobb sorozat ez év vége felé kerülne a piacra. Tulajdonképpen jó lenne az igényeket ismerni, ezért kötelezettség nélkül várjuk a jelentkezéseket. A számítógép megépítéséről szóló cikksorozatot júniusban megindítottuk.

Gilicz András bányamérnök, Nagykanizsa,

Béke u. 14. 8800

Örömmel tallóztam újonnan megjelent, színes, változatos μ Magazinjukban, és azt is örömmel tapasztaltam, hogy a lap lándzsát tör a manapság sokat emlegetett számítástechnikai oktatás és szakemberképzés mellett. Mindazonáltal úgy érzem, hogy az utóbbi gond felvetődésekor a figyelem elsősorban a középiskolás diákok és a leendő, vagy már végzett középiskolás tanárok felé fordul, és a meglevő, más területeken dolgozó szakemberek át- és továbbképzése mintha valamelyest háttérbe szorulna; számukra a lehetőségek száma talán kevesebb. Tudom, szép számban vannak hosszabb-rövidebb, ilyen célú számítástechnikai tanfolyamok, de ezekkel elmélyültebb tudás aligha szerezhető.

Hogy konkrét példával éljek, saját esetemet hozom fel: végzett bányamérnök vagyok, jómagam is számítógépen (TPA–1140) dolgozom, lelkes BASIC és FORTRAN programozó vagyok, a középiskolában és az egyetemen is mindig jó jegyeim voltak matematikából, bújom a hazai és a külföldi, számítástechnikával kapcsolatos publikációkat, mégis úgy érzem, hogy elmélyültebb, rendszeresebb tanulmányokra lenne szükségem. Tanfolyamokon ezt megszerezni nem lehet, erre csak valamelyik felsőoktatási intézmény berkeiben volna mód. A Felvételi Tájékoztató című kiadványt már évek óta figyelem, de ott továbbtanulási lehetőség – például programozó vagy programtervező matematikusi végzettség megszerzésére – tudomásom szerint csak esti tagozaton, Budapesten (ELTE) van, ami számomra – vidéki lévén – elérhetetlen.

Várható-e, hogy a közeli jövőben ez megváltozik, és bevezetik a levelező tagozaton történő számítástechnikai, alkalmazott matematikusi oktatást, ami lényegesen kedvezőbb feltételeket teremtene a vidéki mérnökök számára?

Ha igaz, akkor az MTV, a SZÁMALK és az NJSZT a televízióban BASIC tanfolyamot indít. Kiadunk programozott tankönyvet, megszervezzük a személyi számítógépekkel felszerelt μ klubok rendszerét, ahol a jelentkezők megoldhatják az egyes adásokhoz tartozó feladatokat. A végén a hallgatók vizsgát tehetnek. A részletes felhívást a közeljövőben adjuk ki. Ez volna a kezdet. A továbbiakon sokan gondolkoznak.

Haraszin Tibor, Miskolc I.,

László Jenő u. 94. 3525

Az idén végeztem Automatizálási Főiskolán Kazincbarcikán. Itt volt alkalmam közelebbről megismerkedni a számítástechnikával. Kb. egy-

másfél évig programoztam Intel-Assembler gépi kód szinten. A munkahelyemen közvetlenül nem volt alkalmam számítógépekkel foglalkozni, de egy másik osztályon „kegyelemből" megengedték, hogy munkaidő után gyötörhessek egy Commodore–64-est. Így ismerkedtem meg a BASIC nyelvvel. Ez az öröm azonban csak két hétig tartott, mert megkaptam a behívómat, és azóta sorkatonai szolgálatomat töltöm.

Most jönne a kérésem. Hallottam valami olyat, hogy külföldről vámmentesen hozhatok át számítógépet a saját részemre, amennyiben a munkahelyemről igazolást tudok felmutatni, hogy a gép szükséges munkaköröm ellátásához, valamint aláírom, hogy a gépet 5 évig nem adom el. Önöktől kérdezem, hogy van-e ilyen lehetőség, és ha van, pontosan milyen módon, más is áthozhatja-e a gépet részemre stb.

Elsősorban azért érdeklődöm így, mert kezdő mérnöki fizetésem nem teszi lehetővé, hogy saját gépet vásároljak a hazai kínálatból.

A sorkatonák részére szervezett μ klubokról, illetve a kit számítógépről már szóltam. Az Ötlet 1984. márciusi száma foglalkozik a számítógépek vámkezelésével. Majdnem az a helyzet, amit írt – csak egy kicsit más.

Mayer István János, Pápa,

Béke u. 6. 8500

Körülbelül 2 éve tanultam meg a BASIC nyelvet, és azóta gyakorlom iskolai és magántulajdonban levő gépeken (HT–1080Z, ZX81, Commodore VC–20). Szívesen írok programokat a lapnak bemutatásra, vagy cikket bizonyos témákról, ha tudok segíteni ezzel a szerkesztésben. Különben pedig negyedik gimnáziumba járok. Minden szabad időmet gép mellett töltöm. Úgy is mondhatom: mániám lett. Programozónak szeretnék tanulni is. Bemutatkozó programomat következő levelemben küldöm, de előbb le kell írni, mert csak szalagon van meg egészben minden programom.

Amit leír, dicséretes. Egyetlen veszélye van az ügynek, el ne felejtse, hogy a középiskolában nemcsak számítástechnikát kell tanulni. Ha a többi tárgyat elhanyagolja, számítástechnikai jövőjét is veszélyezteti. A programot csak küldje. Ha mágnesszalagot kapunk, akkor ki fogjuk listáztatni!

Székelyhidi Tamás, Ercsi,

Sallai u. 4. 2451

Levelére a választ már elküldtem. Egy részletét azért közöljük, mert tanulságos.

Fiatal általános iskolai matematika–fizika tanár vagyok, és már régóta érdekel a programozás, ha nem is üzöm profi szinten. Az elmúlt nyár nagy részét módom volt együtt tölteni egy HT–1080Z géppel, és mondhatom, kellemes társnak bizonyult. Természetesen azonnal elkezdett motoszkálni bennem a gondolat, hogy valami módon ezt az örömet meg kell osztani tanítványaimmal is.

Látva, hallva a középiskolás tanulók örömét, szakértelmét, szeretnék feltétlenül találni valamilyen lehetőséget arra, hogy hozzáférhetővé tegyem minden újra fogékony általános iskolás tanulóink számára ezt a varázslatos, játékosan komoly tudományt. Kissé megkeseredik a szám íze, hallva a panaszokat a gépek számának korlátozott voltáról. (Hej, ha egy, csak egy...) Talán ez az „irigység" megbocsátható.

Az iskolaszámítógép-programnak az általános iskolákra való kiterjesztése hamarosan várható. A többi kérdésére a fenti levelekre adott válaszokban talál feleletet.

Takács László, Kerepestarcsa,

Árpád u. 2. 2143

Már éppen ideje volna, hogy egy hardveres szaklap is megjelenjen. Most végzek Székesfehérváron a Számítógéptechnikai Intézetnél, és tapasztalatból tudom, hogy a legközönségesebb, LSI, VLSI áramkörök részletes működésének megismerése is komoly gond volt, nem beszélve nagyobb egységekről. Műszaki könyvtáram kb. 1500 kötet, ebből mintegy 200 számítástechnikai, mégis jelentős hiányt érzek.

Az LSI ATSZ Zilog családjáról megjelent kiadványban is találtam hibákat. Saját gépemen mindjárt kipróbáltam (Z80-as, assemblerje felismeri a nem publikált kódokat is). A számítástechnika nálunk közügy lett, de a nyilatkozatokon kívül általános tájékoztató irodalom alig van. A Rádiótechnika néhány befejezetlen cikke gyakorlatilag zéro. 1963-tól megvan nekem két cseh lap minden példánya, és lényegesen több van bennük, mint a magyar lapokban. Például 2 éve jelent meg egy 8080 alapú gép egy BASIC programmal, szoftver DUMP táblával, leírással.

A μ M szeretné mindezt, amiről ír, pótolni. Értesítsen majd, hogy sikerült-e!

Leveleiket továbbra is várja

KOVÁCS GYŐZŐ

A 4. Számítógép Sakkvilágbajnokság

A budapesti VB befejezését követően, október 22-től 25-ig tartott a 22 résztvevővel, ötfordulós, svájci rendszerben megtartott „nagy világbajnokság". Ez a verseny mind kevésbé tisztán nagygépekre írt programok vetélkedője; ez alkalommal a résztvevők sorában 9 mikroszámítógép szerepelt, közülük 4 a budapesti versenyről utazott – a programozókkal együtt – egyenesen New Yorkba.

Az eredmény

Nagy meglepetést jelentett, hogy a mindeddig világelső, hírneves Belle két vereséget is szenvedett, és a 6.–8. hellyel kellett megelégednie. A világbajnokságot Cray Blitz nyerte meg, 4,5 ponttal; Bebe és Awit 4, Nuchess és Chaos 3,5 pontot szerzett. Közülük Bebe mikroszámítógép. Igaz, speciálisan kialakított processzorral működik, ami a kifejezetten mikrók részére kiírt versenyeken nem megengedett. A Budapesten is szerepelt mikrók közül kiválóan játszott Mephisto X, amely 3 pontot gyűjtött, ugyanannyit mint Belle! Fidelity X (a Prestige – Elite programmal) és Novag X szintén kitűnő, 50 százalékos eredményt ért el, 2,5 pontot. Megállapítható tehát, hogy a sakk-mikroszámítógépek ma már a nagygépekre írt programok között is helytállnak, ami rendkívül örvendetes, hiszen a nagyközönséghez mégis közelebb állnak.

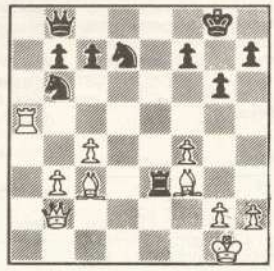
Mintegy találomra emelünk ki két érdekes hadállást a New York-i világbajnokságon váltott játszmák közül.

Pontos számítás

Mephisto sikere egyben a B stratégiáé, különösen ha meggondoljuk, hogy a nagygépekre írt programok java része – kihasználva azok sokszorta nagyobb sebességét – a „brute force"-on alapszik. Bemutatott játszmájában igen régi, tudományos alapon kidolgozott program volt az ellenfele, amely legelső formájában már az 1974. évi világbajnokságon részt vett.

Alkotója, Monroe Newborn kanadai professzor a számítógépes sakkozás egyik legkiválóbb művelője; most választották az ICCA elnökévé, a 7 esztendő után e funkciójából visszavonult Mittman professzor helyébe. Mephisto kifogástalanul vezetett támadással, és igen alapos számítást igénylő lebonyolítással győzött.

Mephisto X – Ostrich



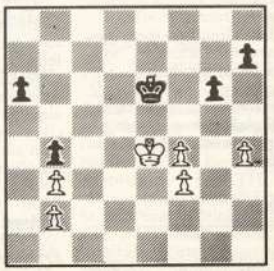
Világos lép

A fenti állásban következett *24. Vd2!* (Fh8-ra Vf8 védi a mattot.) *24. – Be7 25. Vd4* (Most Vg7 és Vh8 matt is fenyeget.) *25. – f6 26. c5, Hc8 27. Fg4, b6?* (Kézenfekvő lépés, mégis hiba. A következő lépéstől a 32-ig világosnak mindent végig kellett számolnia.) *28. Vd5+, Kg7 29. Ba8, c6* (egyetlen lépés) *30. Bxb8, cxd5 31. Fxd7, Bxd7 32. Bxc8, bxc5 33. Bxc5* és világos tiszt előnyét érvényesítve győzött. Mesteri játékvezetés.

Oppozíció

A gyalogvégjátékokban így nevezik a királyok egymással való szembenállását, amelyet ha az egyik fél kénytelen feladni, a másik behatol hadállásába és győz. A legegyszerűbb példáját láthattuk ennek a 2. sz. hadállásban, 1. hxg5, hxg5 2. g4, Kh5 4. Kf2, Kxg4 5. Kg2-vel világos tartja az oppozíciót, és a játszma döntetlen. Ha 3. Kg2??-t lép, akkor Kg4 4. Kf2, Kh3-mal elveszti az oppozíciót és a játszmát is. Az oppozíció kényszerű feladásának esett áldozatul Belle is.

Nuchess – Belle



Világos lép

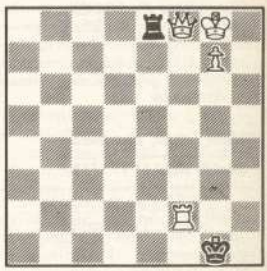
Világos állása látszólag kedvezőtlen, mert 1. Kd3, Kf5 2. Ke3, a5 3. Ke2, Kxf4 4. Kf2, h5 után kénytelen feladni az oppozíciót, és a sötét király behatolva a világos hadállásba, győz. De megfordította a helyzetet: *1. f5+!, gxf5 2. Kd4, Kd6 3. f4!, Ke6 4. Kc5, a5 5. h5, Kf7 6. Kd5* és világos nyert. Ez a nyerés fejlett játéktudást igényelt.

A nagyszámítógép sebessége

A mikrók szép teljesítménye a nagyok között akkor értékelhető igazán, ha képet alkotunk arról az óriási különbségről, amely sebességük között fennáll. Műszaki adatok helyett érdekes példával illusztráljuk ezt. A következő feladványt közöltük az „... és játék" című folyóirat 1983/3. számában, amelynek angol nyelvű példányait – főként e sorok írójának a mai mikroszámítógépekről írt cikkére való tekintettel – New Yorkban is terjesztettük.

S. Clausen

Die Schwalbe, 1939. I. díj



Matt 3 lépésben

Részben az irodalomban, nagyobb részt pedig saját tesztjeink alapján tudjuk, hogy a különböző mikroszámítógépek, amelyek a legutóbbi három évben jelentek meg, mennyi idő alatt fejtették meg ezt a feladványt. A következő felsorolás a sok különféle gép fejtési idejét – kb. piacra kerülésük sorrendjében – tartalmazza. A kulcslépés egyébként *1. Ba2!!*

Sargon Multigame System (Applied)	**1 p 31,2 mp**
Chess Challenger Voice (Fidelity)	**29 p 26,0 mp**
Chess Champion Super System III. (Novag)	**4 p 2,5 mp**
Mephisto I. (Hegener + Glaser)	**1 p 53,1 mp**
Chess Challenger Sensory Voice (Fidelity)	**21 p 23,8 mp**
Sandy Encore (Applied)	**55,8 mp**
Multigame System (Applied)	**53,8 mp**
Multigame System, Capablanca modul (Applied)	**11,0 mp**
Multigame System, Morphy modul (Applied)	**56,0 mp**
Chess Challenger Champion Sensory (Fidelity)	**18,0 mp**
Mephisto II. (Hegener + Glaser)	**1 p 40,0 mp**
Mark V. (Scisys)	**6,0 mp**
Chess Challenger Sensory 9 (Fidelity)	**1 p 36,0 mp**
Mephisto III. (Hegener + Glaser)	**1 p 5,0 mp**
Elite (Fidelity)	**16,0 mp**
Constellation (Novag)	**9,0 mp**

A táblázat sok program jellegzetességére rámutat: a Mark V. fejtés-orientáltságára, a Mephisto mély számításai miatti viszonylagos lassúságára, sőt arra is, hogy a Fidelity számítógépek programjai

H. Berliner (Patsoc 2,0) és A. Marsland (Awit) versenyez

azóta jók, amióta Spracklenék készítik őket a cég részére. Külön érdekesség, hogy az MGS modulok közül a végjátékra specializált Capablanca találja meg jóval gyorsabban a mattot, mint a középjátékra szánt Morphy. Bámulatos a Constellation matt-orientáltsága, amelyet a tévében látott teszt is tanúsított.

Mindezeknél érdekesebb azonban egy levél, amelyet Harry L. Nelsontól, a New Yorkban világbajnokságot nyert Cray Blitz egyik programozójától kaptunk. „Kíváncsiak voltunk, mennyi időre van számítógépünknek szüksége e feladvány megfejtéséhez. A Cray–1 számítógépet használva éppen 0,5 mp-nél kicsivel több kellett. A VB-n azonban a Cray–XMP–2-t használtuk (ma a világ leggyorsabb és legnagyobb számítógépének tekinthető. – A szerk.), amely kétszer olyan gyors. Merem tehát mondani, hogy azzal a géppel 0,5 mp sem kellett volna hozzá."

Ilyen számítógépekkel versenyeznek napjainkban a derék mikrók.

DR. LINDNER LÁSZLÓ

A sakkprogramozási pályázat hírei

Az ez évi első számunkban kiírt pályázat díjalapja – az eddig beérkezett felajánlások szerint – megközelíti a százezer forintot. Ezt az összeget a zsűri a pályázatok száma, értéke és jellege szerint, esetleg több csoportba osztja fel. A díjalap végleges, pontos összegét még a pályázat beküldési határideje előtt közöljük.

A részt venni szándékozók segítésére a Neumann János Számítógéptudományi Társaság Budapesten, a területi szervezetek és klubok vidéken – lehetőségeikhez képest – hozzásegítenek számítógépek használatához. Az érdeklődők sürgősen jelentsék be ez irányú igényeiket. A szervezők felkérik a vállalatokat, közöljék, ha erre a célra szabad gépidejük van. Kérik azokat is, akik – például munkahelyükön vagy saját számítógépük segítségével – önerőből kívánnak részt venni a pályázaton, szándékukat ugyancsak közöljék szerkesztőségünkkel.

A szervező bizottságok – írásbeli megkeresésre – szívesen adnak felvilágosítást, szaktanácsot.

A pályázat határidejét 1985. június 30-ig meghosszabbították.

Új 16 bites mikroprocesszor

Az NDK-ban U 880 néven már 1980 óta gyártják a Zilog cég Z80 mikroprocesszorának funkcionális megfelelőjét. Az ezzel kapcsolatos fejlesztések most minőségi előrelépést eredményeztek. A tavaszi Lipcsei Vásár egyik szenzációjaként mutatkozott be a 16 bites U 8002 mikroprocesszor, amely a Zilog Z 8002-jének funkcionális megfelelője. A vásáron kapott tájékoztatás szerint a Z 8002 sorozatgyártása 1985-ben kezdődik.

Új hullám jövőre

Az IBM PC az 1982-es USA-beli, majd az azt követő 1983-as európai bejelentés után szerte a világon egyre gyorsabb ütemben terjed. Éppen ezért nagy jelentősége van az egyes géptípusok forgalmazhatóságának, hogy azok kompatibilisak-e az IBM PC-vel.

A szocialista országokba is „begyűrűzött" ez a tendencia. Elsőként már tavaly tavasszal a hazai SZKI reagált, piacra hozva a PROPER 16 nevű gépét. Éppen egy évre rá megjelent a második ilyen szocialista gyártmányú gép, amely szintén hazai termék: a VT 16, a VIDEOTON gyártmánya. A becslések szerint legkésőbb jövőre már valamennyi szocialista országban megkezdődik az IBM PC-vel kompatibilis mikroszámítógépek gyártása.

Sakkozógép

Az NDK Mikroelektronikai Kombinátjának erfurti gyára – a szocialista országok közül elsőként – megkezdte a sakk-számítógépek gyártását. Az SC 2 néven készülő gép játékerőssége 10 fokozatban állítható. A mikroprocesszora U 880 (a Z80-nak az NDK-ban gyártott funkcionális megfelelője). A májusban bemutatkozott SC 2-ből a KONSUMEX Külkereskedelmi Vállalat – piacfelmérési célból – 100 darabot már meg is rendelt. Az SC 2 hazai fogyasztói ára kb. 4500 forint.

Házi számítógépek Csehszlovákiában

Az idén már 500 darabot gyártanak a Novij Bor-i ZPA gyárban az IQ 150 típusú házi számítógépből. A 18 ezer koronába kerülő gépet a CVUVT gyárban fejlesztették ki, és kizárólag a szocialista országokban készült alkatrészekből épült fel. Tervezik iskolákban való széles körű terjesztését is.

Az idei Tavaszi Lipcsei Vásáron mutatkozott be először a PMD 85 típusú, Tesla Piestanyban gyártott, a Hewlett Packard cégnek a HP 85 típusával kompatibilis mikroszámítógépe. A max. 48 kbájt operatív tárral rendelkező gép mozgó billentyűzetének érdekessége, hogy 11 funkcionális billentyűje is van.

Hordozható mikroszámítógép

Elkészült a szocialista országok első hordozható mikroszámítógépe. A Műszertechnika GMK által százas sorozatban gyártott elegáns gépet decemberben mutatták be először a szakemberek előtt MINICOMP C–1 néven. Az időközben TRANSMIC-nak átkeresztelt gép a tavaszi BNV-n mutatkozott be először a nagyközönség előtt. A TRANSMIC a professzionális kategóriába tartozik, erre utal a két beépített mini hajlékonylemezes tárolója. Az amerikai Radio Shack TRS–80 típusú gépével kompatibilis. Háromféle operációs rendszert szállítanak hozzá: TRSDOS, NEWDOS és CP/M 2.2 változata.

A Z80 mikroprocesszoros, max. 512 kbájtos operatív tárral rendelkező gép elvileg táskaként hordozható, gyakorlatilag azonban nem, hiszen működtetéséhez feltétlenül szükséges a 220 V-os hálózathoz való csatlakoztatása.

Jön a GKS!

Az ISO már nemzetközi szabványjavaslatként ajánlja a GKS (Grafical Kernel System) nevű grafikus rendszert, mely a szakemberek szerint várhatóan már az idén szabványnyá lép elő. A GKS fő jellegzetessége, hogy a képek eszköz-függetlenül adhatók meg.

Hazánkban az MTA SZTAKI – a szocialista országok közül elsőként – már implementálta a GKS rendszert bármely UNIX-szerű operációs rendszer alá. A BNV-n például már futott a GKS a VIDEOTON új, rendkívül perspektív termékén, a VT 32-n, az UNIX-szerű SOS operációs rendszer vezérlete alatt. Az MTA SZTAKI GKS-implementálása iránt nyugaton is igen nagy az érdeklődés, például a világon a legtöbb terminált értékesített TEKTRONIX cég is már tárgyalásokat kezdeményezett.

Háztartási tévé, magnó + hajlékonylemez

MERIUM néven modulárisan kiépíthető mikroszámítógépet kezdtek gyártani a lengyelországi Zabrze-ben, a MERA gyárában. A személyi számítógép kategóriába tartozó géphez csatlakoztatható két hajlékonylemezes tároló, háztartási színes tévé és háztartási magnetofon is. A soros illesztője az RS232C-nek megfelelő. A Z80-nak az NDK-ban gyártott funkcionális megfelelőjén az U 880 típusú mikroprocesszort tartalmazó gép operatív memóriája 18 kbájtos. A 14 kbájtos, csak olvasható tárban található BASIC fordítónak az értelmező változata. Ez kompatibilis az amerikai Radio Shack cég TRS 80–1 típusú gépén futó BASIC nyelvvel.

A CP/M előretör

A 8-bites gépeknél az alkalmazott operációs rendszert tekintve már a szocialista országoknál is megfigyelhető a lassan világszabvánnyá váló CP/M felé törekvés. A szocialista gyártmányú 8-bites mikroszámítógépeknek már a 20%-a használ CP/M vagy CP/M kompatibilis operációs rendszert. E tendencia még markánsabban megfigyelhető, ha hozzátesszük, hogy az 1983-ban és az 1984-es év elején a szocialista országokban megjelent, nagy sorozatú gyártásra tervezett professzionális bázisgépek mindegyike rendelkezik CP/M kompatibilis operációs rendszerrel is. Például a bolgár IZOT 103.1 (1984), a csehszlovák SZM 50/40 (1983), a lengyel ELWRO 500 (1983), a magyar PROPER 8 (1983) és VT 16 (1984), a román CUB (1983), a szovjet SZM 1800 (1983).

A nagydíjas

A tavaszi BNV-n az elektronikai és számítástechnikai termékek közül a Számítástechnikai Koordinációs Intézet PROPER 16A professzionális mikroszámítógépe kapott nagydíjat. A tavalyi tavaszi BNV-n bemutatkozott PROPER családnak ez már egy korszerűsített tagja. Az Intel 8088 típusú kvázi-16 bites mikroprocesszora kompatibilitást biztosít a PROPOS operációs rendszerén keresztül a szerte a világon rendkívül gyorsan terjedő IBM PC-vel. A PROPER 16A már az idén százas sorozatban készül.

A Műszertechnika GMK 1981-ben alakult az első PJT-k között. 1982–83-ban négyféle műszert és ötféle különböző kategóriájú számítógépet fejlesztett ki. A kifejlesztett számítógépeiből két típusnak a gyártási jogát eladta. A GMK forgalmának állandó növekedése (1984. évi tervezett árbevétele 50–80 millió forint) lehetővé tette a profiltisztítást, csak a 8, 16 bites számítógépek, illetve számítógéphálózatok fejlesztését és gyártását. 1984-ben csak a 8 bites számítógépeiből kb. 150 darabot gyárt. A 16 bites számítógépek (M 68000 bázisúak) 1985-től már forintért is kaphatók lesznek. Mind a 8 bites (Z80 bázisú), mind a 16 bites számítógépeit asztali és hordozható kivitelben is gyártja.

A Műszertechnika részlegei:
hardverfejlesztés
elektromos gyártás
mechanikus gyártás
rendszerszervezés és szoftverfejlesztés.

A részlegek hatékony együttműködése lehetővé tette a komplett, kulcsrakész rendszerek szállítását. A Műszertechnika vállalja az alábbi komplett rendszerfajták szállítását:
mérő- és folyamatirányító rendszerek
szövegfeldolgozó munkahelyek
adatfeldolgozó
(készletgazdálkodás, raktári nyilvántartás,
számviteli feladatok,
idő-, munkaerő-nyilvántartás,
bérszámfejtés,
rendelésnyilvántartás stb.) rendszerek
hálózatos szoftverfejlesztések

Hordozható professzionális személyi számítógép

TRANSMIC

központi egység: Z80 bázisú
operatív memória: 64 kbájt
(256 kbájtig bővíthető)
2 db 256 kbájt kapacitású mini floppy
(a későbbiekben 512 kbájtra bővíthető
zöldcsövű display
párhuzamos interfész (Centronics típusú)
nyomtató felé soros interfész (RS–232 C)

használható operációs rendszerek:
NEWDOS (Radio Shack programok futtatására)
CP/M 2.2 (IBM programok futtatására)
TRSDOS
MSYS

nagy mennyiségű felhasználói szoftver

opcionális lehetőségek:
multiplexer, A/D konverter
D/A konverter

intelligens terminálként való használat
akkumulátoros üzemmód

fejlesztés alatt: Winchester illesztő
M 68000 (16 bites) opcionális kártya

A számítógépek garanciális ideje 1 év.

Utána 10 évig térítés ellenében szerviz.

Szállítási határidő:
kulcsrakész rendszerek esetén negyedév
különösen kiemelt feladatoknál 1 hónap

A gép árában két fő oktatása, illetve kiképzése benne van. Kulcsrakész rendszerek szállítása esetén a számítógépek bérelhetők.

Címünk: **Műszertechnika GMK**
1126 Budapest
Kiss János altb. u. 26.
Telefon: 363-623
222-069 (üzenetrögzítővel ellátva)

**Kovács Imre–
Szabó Gizella–
Tick József–
Tiszai Tamás:
ABC–80 kezdőknek
(Bp. 1982.
Tudományszervezési
és Informatikai Intézet,
Iskolaszámítógép-sorozat.
128 oldal)**

A várakozás izgalmával vettem kezembe az iskolaszámítógép-sorozat régen várt és hiánypótló, első kötetét, amelynek nem hivalkodó, ízléses kiállítása, a kézikönyvnek megfelelő mérete, tartalomjegyzéke tovább növelte érdeklődésemet. A könyv olvasása közben ez az érzés előbb csodálkozássá, majd döbbenetté, végül felháborodássá változott. Kár volt a Tudományszervezési és Informatikai Intézetnek (a könyv kiadójának) a számítástechnika iránt anynyira fogékony és érdeklődő fiatalok kezébe ennyire pongyolán fogalmazott, a számítástechnikai szakkifejezéseket a szakmában meghonosodott jelentéstől eltérően használó könyvet adnia!

A bevezetés elméleti ismereteket és gyakorlati tanácsokat ígér. Ezek használhatóságát azonban pontatlanságaik nagymértékben nehezítik. Néhány kiragadott – sajnos nem kevés – tévedésből, pontatlanságból:

„A BASIC utasítások általános felépítése:

címke utasításkulcsszó operandus sorvégjel"

A felírásban, de a hozzá tartozó magyarázatban is keveredik az utasítás és a programsor, a címke és a sorszám, a RETURN billentyű és a sorvégjel fogalma.

A gyakorlati felhasználást az nehezíti, hogy a szerzőknek végig nem sikerült eldönteniük, kiknek – kezdőknek, haladóknak, profiknak – írják ezt a könyvet. A példaanyag aránytalanságait jól mutatja például az, hogy a leghosszabb példa a CHAIN utasítás használatát mutatja be. Ezt az utasítást a kezdők a legritkább esetben használják.

Tovább nehezíti a felhasználást, hogy az utasítások hatásának leírásában néhány helyen pongyola a megfogalmazás. Kiragadott példák: 23. oldal: „FIX(X) az aritmetikai kifejezésben bennfoglalt legnagyobb egész szám" (????). 124. oldal: „MID ¤ (kar-kif, arit-kif, arit-kif) – A karaktersorozat középső része." Találja ki az olvasó, hogy a két arit-kif hogyan is definiálja a karatersorozat közepét.

Bizonyos, kézikönyvben kötelező részek hiányoznak. Ilyen például a kiírást vezérlő karakterek táblázata, a memóriatérkép, a busz vonalainak leírása, a lemezegység leírása, a RESET kapcsoló feladatának leírása stb. Nem szól az ABC–80 használata közben felmerülő problémákról, szoftverhibákról.

A belső borítón hiába keressük a lektor nevét; úgy tűnik, a kézikönyvet senki sem lektorálta. Pedig egy alapos szakmai ellenőrzés mindenképpen megszüntette volna ezeket az aránytalanságokat és pontatlanságokat.

Nagy szükség lett volna szakmai nyelvi lektorra is. A könyv igen széles kör kezébe kerül, ezért mindenképpen ügyelni kellett volna arra, hogy az általánosan elfogadott terminológiát, és ne a szerzők zsargonját használja.

A könyvnek természetesen vannak élvezetes és pontos részei is. Kár, hogy ezek éppen azok, amelyek az ABC–80 egyszeri használata után elsajátíthatók (például a billentyűzet használata).

A könyv igazán értékes része a Függelék, amely összefoglalja a BASIC szintaxisát és a hibaüzeneteket.

–yl–

•

A rovatszerkesztő megjegyzése: A könyvbeli hibák jó része a korai – akkor még úttörő – megjelenéssel magyarázható.

**Herneczky Katalin–
Orbán Katalin:
A JACKSON-féle
programtervezési módszer
(Bp. 1982.
Tudományszervezési
és Informatikai Intézet,
286 oldal. Ára: 130,– Ft)**

A kiadvány az első, önálló használatra alkalmas, magyar nyelvű munka M. Jackson programtervezési elveiről, és mint ilyen, hiánypótló a magyar számítástechnikai irodalomban. Betekintést nyújt az M. Jackson által kifejlesztett, programnyelvtől független programtervezési módszerbe. A szerzők könnyen érthető, röviden bemutatható, de egyben megfelelő általánosítások levonására is alkalmas példákon jelenítik meg a módszert.

A könyv bevezető részében összefoglalást nyújt a modern strukturált programkészítés módszereiről és elméleti alapjairól. Ismerteti a Böhm–Jacopini tételt, a Mills-féle komplex struktúrát, Dijkstra hierarchikus programozási módszerét, a Wirth-féle lépésenkénti finomítás elvét, és áttekintést ad a Jackson-módszer alapjairól: a tervezés lépéseiről, az alapszerkezetek struktúra-diagramjairól, az alapszerkezetek kódolásáról.

A továbbiakban a könyv jellegzetes adatfeldolgozási problémákkal – összegfokozatos, összeválogatásos (merge) és naprakészítési (update) feladatokkal – foglalkozik, egyidejűleg strukturált PL/1 és COBOL kódolási konvenciókat tekint át és alkalmaz feladatok megoldásánál.

A visszaléptetéses (backtracking) technika ismertetése során a PL/1 és COBOL kódolási konvenciók használata mellett a szerzők részletes feladatmegoldás keretében foglalkoznak a kétszeres pufferezéssel és az egymásba ágyazott posit-admit szerkezettel.

A könyv befejező része „A struktúra-ütközés és a programinverzió". Ez a fejezet a struktúra-ütközés három fajtáját vizsgálja: a rendezési ütközést (ordering clash), a tagolási ütközést (boundary clash) és az összefonódási ütközést (multi-threading clash). Mindhárom esetben feladatmegoldás szolgálja a jobb megértést.

A kiadvány végén szakirodalmi összeállítás található.

A könyvet elsősorban azoknak ajánljuk, akik megfelelő számítástechnikai ismeretekkel rendelkeznek és ismernek legalább egy magas szintű programnyelvet.

DR. NÉMETH GYÖRGY

NEMZETKÖZI SZÁMÍTÁSTECHNIKA

Az Internacia Komputado (Nemzetközi Számítástechnika) kiadvány-sorozat a számítástechnika tömegesen terjedő alkalmazásával foglalkozik, népszerű módon, magazin jellegű formában.

Az elsődleges célja, hogy javítsa a számítástechnikáról, mint az emberiség kultúrájának szerves részéről szóló információk nemzetközi áramlását. Ennek érdekében különös figyelmet szentel a kis országoknak, a fejlődő országoknak és a nyelvileg elszigetelt országoknak. Erre a nemes és felelősségteljes feladatra az Internacia Komputado reális esélyekkel vállalkozik: a világ csaknem száz országában való kedvező fogadtatásának titka nyelvének abban a semlegességében rejlik, amit az eszperantó biztosít számára.

Az Internacia Komputadonak vállalt feladata ellátásához tömegkapcsolatokra van szüksége. Ezt szolgálja egyrészt az, hogy a világ több tucat országában előfizethető nemzeti valutában, másrészt pedig az előfizetési díj – igazodva az egyes országokban élők jövedelméhez – mindenütt magánszemély által is könnyen kifizethető, reális összeg.

Igen sokrétű az Internacia Komputado-nak a mind az öt kontinenst felölelő előfizetői tábora: az általános iskolás kisdiáktól kezdve egészen a nemzeti tudományos akadémia számítástudományi intézetéig, sőt oktatási minisztériumig is terjed.

Az eddig megjelent kiadványo témái a Mikroszámítógépek, a Hálózatok és a Szövegfeldolgozás. Ezek egyenként is megvásárolhatók 30 forintos áron a Statisztikai Kiadó Vállalat Könyvesboltjában (Budapest II., Keleti Károly u. 10.). A sorozat elő is fizethető, négy számért 100 forintot befizetve az OTP VII. ker. 218-98330/514-15629 számú számlára MESZ Számítástechnika 1368 Budapest, pf. 193. címre, melyen postautalvány is igényelhető.

MIKI

TENSITORQUE

Forgatónyomték-, fordulatszám- és teljesítménymérő műszercsalád

Felhasználási terület:

A *forgatónyomaték* és a forgó tengelyeken átvitt *teljesítmény* mérése fontos feladat, többek között

- a gépek és gépelemek szilárdsági méretezése
- az erő- és munkagépek működési hatásfokának megállapítása
- az üzemi paraméterek mérése és
- diagnosztikai vizsgálatok céljából.

A témakör nagy jelentőségű az anyag- és energiatakarékosság szempontjából egyaránt.

A MIKI Méréstechnikai Fejlesztő Vállalatnál kifejlesztett széles méréstartomány lehetővé teszi a legkisebb egységekkel az elektronikus áramkörökben alkalmazott miniatűr forgókapcsolók nyomatékszükségletének mérését, a legmagasabb méréstartományokban a közúti és mezőgazdasági nehézjárművek fejlesztéséhez és ellenőrző vizsgálataihoz a maximálisan 20 kNm-es nyomatékok mérését.
Az érzékelőkhöz különféle kijelző, regisztráló és adatfeldolgozó műszerek csatlakoznak. Az egyik legbővebb kiépítés a Tensitorque–1 nevű futómű élettartam vizsgáló számítógépes mérőrendszer.

Ismertetés:

A mérőfej precíziós csapágyazású, speciális anyagból készült mérőtengelyt tartalmaz, melynek alkalmasan kiképzett felületére nagyállékonyságú nyúlásmérő-bélyegeket ragasztunk fel. A mérőbélyegekből kialakított Wheatstone-hidat a magasabb méréshatárokban csúszóérintkező nélkül, forgó transzformátorral tápláljuk, és a híd kimenő feszültségét, mely a mérendő nyomatékkal arányos, a tengely furatában mérőpatronban elhelyezett elektronika segítségével ugyancsak csúszóérintkező nélkül juttatjuk ki az állórészbe. A mérőfej optikai fordulatszám-érzékelőt is tartalmaz, így a nyomaték és a fordulatszám egyidejű mérésével lehetővé teszi a tengelyen átvitt teljesítmény mérését.

Műszaki adatok:

Méréshatárok:	
Forgatónyomaték:	0,2 Nm…20 kNm az 1., 2., 5. sor szerinti bontásban
Fordulatszám-mérés:	5000 ford/perc 1 kNm-ig 3000 ford/perc 2…10 kNm-ig 1000 ford/perc 20 kNm-nél
Szögelfordulás-mérés:	10 × 360° mérés-határban
Pontosság:	(±/0,2…1%) a végértékre vonatkoztatva
Cellatényező:	1 mV/V (±/0,1…1%)
Tápfeszültség:	6 V egyen vagy váltakozó
Hőkompenzált hőmérséklet-tartomány:	−20… +50 °C
Nullhelyzet és érzékenység hőmérséklet-függése:	(±/0,05…0,1%) 10 °C
Kivitel:	Pormentesen zárt
Az 1 kNm fölötti típusoknál:	
A 0 nyomatékhoz tartozó kimenőjel:	10 kHz
Kimenőjel változás a 0 és ± névleges nyomaték között:	±5 kHz ±0,1%
A ± névleges nyomatékhoz tartozó kimenőjel:	12 V csúcstól csúcsig
Kalibráló jel:	Az adattáblán megadva.

TENSITORQUE–1

Futóműélettartam-vizsgáló számítógépes mérőrendszer

A fentiekben ismertetett érzékelők felhasználásával működő laboratóriumi programozható elektronikus célberendezés, forgatónyomatékok, fordulatszámok mérésére, valamint teljesítmény és hatásfok számítására.

Szolgáltatásai:

- Egyidejűleg 1, 2 vagy 3 érzékelő nyomaték és fordulatszám jeleit fogadja.
- Fogadja egy komplex berendezés hőmérséklet jeleit
- Kiadja a zárt láncú szabályozáshoz szükséges nyomaték, fordulatszám és teljesítmény jeleket
- Működteti a határérték túllépését érzékelő jelfogókat
- Feldolgozza a bemenőjeleket és hatásfok, teljesítmény stb. számításokat végez
- A kapott eredményeket naplózza
- A pillanatnyi eredményeket képernyőn megjeleníteni
- Az operátor beavatkozásait, ill. az eljárás vezérlését billentyűzeten fogadja
- A nyomaték, fordulatszám és teljesítmény kiválasztott érzékelőre vonatkozó pillanatnyi értékeit egy-egy digitális kijelzőre kiírja és határérték túllépés esetén fényjelzést ad
- A kiválasztott tengely elfordulásait számláló mutatja és egy előre beállított értéknél jelzést ad

MIKI MSFv

MIKI, MÉRÉSTECHNIKAI SOFTWARE FEJLESZTŐ LEÁNYVÁLLALAT
1122 BUDAPEST XII.,
PETHÉNYI ÚT 5–7.

Szoftverház méréstechnikai és egyéb speciális szoftver készítésére

- Kisvállalat – gyorsaság – rugalmasság
- Mikroprocesszoros (I 8080, I 8085, Z80, M6800,…) rendszerek, speciális felhasználói programok ipari és fejleszt célokra
- Szoftverrendszerek tervezése, programozása, implementálása, követé
- Programfejlesztés assembly és magasa szintű (BASIC, FORTRAN, PASCAL, CDI stb.) nyelven, interaktív eszközök, számítógépes szimulálás, emulálás
- Firmware-készítés, EPROM (2708, 2716, 2732) égetés, másolás, ellenőrzés

Referencia-rendszerek:

- többprocesszoros rendszer postai PCM hálózat ellenőrzésére, adatok fogadása elemzése, kijelzés képernyőn, protokoll sornyomtatón, operátorprogram;
- adatgyűjtő hálózat húsipari kombinátbar mérleg-terminálok vezérlése, digitális szűrés, vonali programok, központi adatrendezés, kijelzés, archiválás;
- timföldgyári zagybeállító rendszer, anal jelek mérése, számítások, szelepek vezérlése, kijelzés képernyőn, protokollnyomtatón;
- tejüzemi technológia vezérlése, sématábla, operátorpult, folyamatkiválasztás, indítás, logikai ellenőrzések;
- automata mérőrendszer, szerelt kártyák beültetésének ellenőrzése, speciális mérőnyelv, mérőprogram-könyvtár, sorozatmérések kiértékelése.

A műszaki fejlesztés érdekében a változtatás jogát fenntartjuk.

Egyedi berendezések tervezését, kivitelezését, beépítését és betanítását vállaljuk.

Multiprocesszoros mikroszámítógép-rendszer
- ipari folyamatvezérlés,
- adatgyűjtés,
- feldolgozás,
- programfejlesztés

céljaira
(az SZKI M40 XXX kártyarendszer licence alapján)

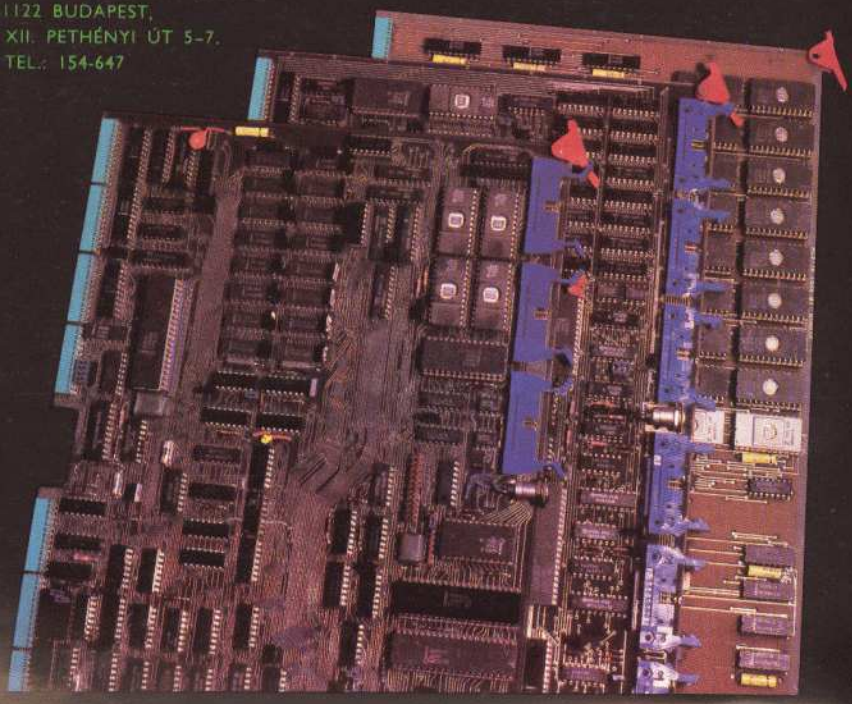
Széles körű szoftvertámogatás:
- CP/M kompatibilis operációs rendszer,
- C-, PASCAL- és assembly nyelvű programfejlesztési lehetőség,
- Keresztfejlesztés RSX–11M operációs rendszer alatt C és assembly nyelven

Megbízható konstrukció, négyrétegű nyomtatott áramköri kártyák
- Intel MULTIBUS-ra épülő I 8085 alapú master/slave kártyák,
- 1 Mbájt memóriatartomány
  - 128 kbájt dyn. RAM kártyák,
  - 64 kbájt EPROM/RAM kártyák,
  - memória rendszervezérlő kártya
- Perifériacsatoló szendvicskártyák
  - 2,5 Mbájt fixfejes MOM diszkcsatoló
  - CM5400 diszkcsatoló
  - Floppy-diszk csatolók,
  - IEC-interfész vezérlő funkciókkal,
  - Modemvezérlő soros interfészek
- 16 bites CPU kártya (iAPX 86/20) (1984. IV. negyedévtől)
  - SBC vagy MULTIBUS-master,
  - 256 kbájt on-board RAM,
  - BSI, RS–232–C, X. 25 interfészek, magas szintű protokollok.

MIKI

MSFV

MIKI. MÉRÉSTECHNIKAI SOFTWARE FEJLESZTŐ LEÁNYVÁLLALAT
1122 BUDAPEST,
XII. PETHÉNYI ÚT 5–7.
TEL.: 154-647



ÁRA: 28 Ft.

Ember
és
számítógép
együttműködéséhez

speciális
szolgáltatások
és szakemberek
szükségesek.

Információs szolgálattal
és
szakmai bemutatóval várjuk az érdeklődőket.

A Miskolci Jubileumi Kiállítás- és Vásáron augusztus 17–26-ig

A SZÜV a számítástechnikai szolgáltatások szakvállalata.

Központi Statisztikai Hivatal Számítástechnikai és Ügyvitelszervező Vállalat